



**Universitat Autònoma
de Barcelona**

TIENDA ONLINE Y GESTOR DE STOCK

Memòria del projecte
d'Enginyeria Tècnica en
Informàtica de **sistemes**
realitzat per
Miguel González Bitrián
i dirigit per
Daniel Blabia Girau

Escola Universitària d'Informàtica
Sabadell, Setembre de 2009

El/la sotasignant, Daniel Blabia Girau
professor/a de l'Escola Universitària d'Informàtica de la UAB,

CERTIFICA:

Que el treball al que correspon la present memòria
ha estat realitzat sota la seva direcció
per en Miguel González Bitrián

I per a que consti firma la present.
Sabadell, Setembre de 2009

Signat:

Resumen del proyecto

El objetivo de este proyecto es desarrollar una aplicación web que sirva y gestione una tienda de música, tanto para su tienda física como para su tienda on-line. Esta aplicación sustituye una metodología insuficiente utilizada durante los últimos años por los responsables de la empresa.

La aplicación Web está gestionada por los usuarios "administrador" y utilizada por los dos tipos de usuarios: administradores y clientes. Sus principales funciones son:

- Introducción y modificación de artículos.
- Gestión de entradas y salidas de productos.
- Gestión de pedidos.
- Obtención de datos para la gestión de la empresa.
- Minimizar los errores de gestión.
- Mejorar la imagen de la empresa.
- Ampliar los ámbitos de negocio.
- Correcta visualización de los artículos.
- Facilitar la búsqueda y compra de artículos.

Esta aplicación se ha desarrollado con las siguientes tecnologías:

- MySQL
- phpMyAdmin
- PHP 5
- HTML
- Javascript
- Ajax
- Excel

Índice

1. INTRODUCCIÓN.....	3
1.1. Motivaciones Personales.....	3
1.2. La Empresa.....	3
1.3. Situación Actual	4
1.4. Clientes.....	4
1.5. Objetivos	5
2. ANÁLISIS UML DEL PROYECTO	6
2.1. Estudio de Viabilidad.....	6
2.1.1. Introducción.....	6
2.1.2. Objeto	6
2.1.3. Sistema a realizar	7
2.1.4. Planificación.....	9
2.1.5. Características del modelo de desarrollo elegido.....	10
2.1.6. Ventajas del modelo de desarrollo elegido	11
2.1.7. Desventajas del modelo de desarrollo elegido	12
2.1.8. Planificación temporal del proyecto	12
2.1.9. Conclusiones	15
2.2. Listado de Requerimientos	15
a) Requerimientos Funcionales.....	15
2.3. Funcionalidades.....	17
2.4. Diagrama casos de uso	18
2.4.1. Clientes	18
2.4.2. Administrador	19
2.5. Casos de uso reales.....	20
2.5.1. Gestor de stock.....	20
2.6. Funcionamiento dinámico de la aplicación	54
3. ANÁLISIS Y DISEÑO DE LA BASE DE DATOS	56
3.1. Diseño Conceptual.....	56
3.1.1. Diagrama Chen.....	56
3.2. Diseño Lógico y Diseño Físico de la Base de Datos.....	58

4. DISEÑO DE LA INTERFICIE GRÁFICA	69
4.1. Gestor de Stock.....	69
4.1.1. Inicio	69
4.1.2. Ventas del día.....	69
4.1.3. Gestor de stock.....	70
4.1.4. Introducir Artículo.....	71
4.1.5. Introducir Compra.....	71
4.1.6. Introducir Sello.....	71
4.1.7. Introducir Distribuidor.....	71
4.1.8. Introducir Estilo	72
4.1.9. Introducir Tamaño	72
4.1.10 Introducir Fabricante	72
4.1.11 Consultar/Modificar Artículo	72
4.1.12 Consultar/Modificar Sello	72
4.1.13 Consultar/Modificar Distribuidor	73
4.1.14 Consultar/Modificar Fabricante.....	73
4.1.15. Eliminar Referencia	73
4.1.16. Eliminar Sello	73
4.1.17. Eliminar Distribuidor.....	73
4.1.18. Eliminar Estilo.....	73
4.1.19. Listados	74
4.2. Web.....	74
5. PRUEBAS.....	76
5.1 Pruebas de caja blanca	77
5.2 Pruebas de caja negra.....	78
5.3 Niveles de prueba	78
5.4 Pruebas realizadas en la aplicación	78
6. OBJETIVOS CONSEGUIDOS	80
6.1. Posibles ampliaciones.....	81
6.1.1. Contratos.....	81
6.1.3. Impresión etiquetas	81
6.1.4. Restock	82
6.1.5. Listado más vendidos	82
6.1.6. Actualización de pedidos.....	82
7. CONCLUSIONES	83
8. BIBLIOGRAFÍA	84

1. INTRODUCCIÓN

El proyecto consiste en el desarrollo de una aplicación Windows/Web que permita llevar el control de stocks de una tienda de música electrónica así como ampliar su mercado y sus ventas. Para poder realizar esto el proyecto se ha dividido en dos partes claramente diferenciadas.

En un principio sólo se iba a realizar la creación de un entorno Web ya que este proyecto se desarrolla en el marco de una tienda existente desde 2004 que en la actualidad cuenta con una presencia mínima en Internet. Una vez iniciado el proyecto se contempló la necesidad de ampliarlo, de manera que fuese más fácil gestionar la tienda así como el poder extraer conclusiones sobre las ventas de la misma, por lo que se decidió ampliar el proyecto creando un gestor de stock.

Hoy en día casi todas las tiendas de este tipo poseen webs muy potentes, capaces de dar mucha información y de ofrecer todo tipo de productos relacionados con la música como tablas de mezclas, patinadores, cascos, platos, bolsas porta vinilos...El proyecto pretende conseguir que la Web se ponga al nivel del mercado actual y que sea capaz de dar toda la información posible tanto al usuario como a los administradores sobre su tienda.

1.1. Motivaciones Personales

La principal motivación a la hora de realizar este proyecto es el hecho de conocer personalmente a los dueños de la empresa, conocer sus proyectos e iniciativas y ayudarles en sus inicios empresariales.

Sabiendo de la necesidad que tenían de gestionar correctamente la tienda y ampliar la empresa me llevó a tomar la decisión de desarrollar una aplicación que les facilitase el trabajo así como la posibilidad de incrementar su cartera de clientes vía Internet. Y de esta manera extraer el máximo partido y ampliar los conocimientos adquiridos durante la carrera en el tratamiento de bases de datos, programación en VisualBasic.Net, desarrollo de páginas Web y el hacer una aplicación desde su inicio hasta su puesta en marcha.

1.2. La Empresa

La empresa a la que está dirigida el proyecto se llama TrivialMusik.net. Es una pequeña empresa de amigos situada en Barcelona, ahora mismo cuenta con 4 trabajadores (incluidos los propietarios) y un pequeño almacén situado en la trastienda. La actividad empresarial de esta empresa se sustenta principalmente de la venta de vinilos y de los porcentajes de representación de artistas.

En el último año el crecimiento del interés por el "Schranz", estilo en el que está especializada la empresa, y del que es pionero en España, ha hecho que las ventas de la tienda y el interés de promotores y discotecas por este estilo musical haya aumentado considerablemente, con el consecuente aumento del trabajo.

1.3. Situación Actual

Actualmente el mercado del "Schranz" está en auge en España por lo que la oferta de este tipo de música (antes despreciado por mucha gente) está creciendo a pasos agigantados.

Hay que decir que el resto de empresas del sector utilizan la misma metodología de trabajo: pedidos vía Internet y compras en sus propias tiendas.

Aunque TrivialMusik es la pionera en España existen otras empresas en el extranjero que poseen una gran parte del mercado debido a que llevan más tiempo en él, y por lo tanto son bastante conocidas. A pesar de esto TrivialMusik se encuentra en un estado que permite hacer frente a este tipo de empresas, en la actualidad se reciben pedidos de países tan lejanos como Brasil o Argentina.

La idea de este proyecto surgió a raíz de los contactos con los propietarios que durante los dos últimos años constataron que tanto las ventas de la tienda como las expectativas de expansión de la empresa han ido creciendo a buen ritmo, hasta llegar al punto de ser bastante difícil el continuar con los métodos de gestión utilizados hasta el momento.

Otras de las motivaciones para la realización de este proyecto han sido los planes de futuro de la empresa. Entre ellos destacan la creación de un sello discográfico propio, la posibilidad de controlar a través de la tienda toda la distribución nacional de *Straight* (distribuidora mundial de música electrónica) y la representación a nivel nacional de un gran número de artistas, tanto nacionales como internacionales con gran proyección. Debido a todo esto la empresa se ha visto en la necesidad de disponer de una aplicación que agilice la gestión de las diferentes líneas de negocio. Así:

- Controlar la entrada y salida de artículos
- El tratamiento de las ventas diarias
- Adaptarse al nuevo negocio de la distribución

Los socios de la empresa constatan además que el grado de conocimiento de su marca es una de las variables a mejorar de cara a sus nuevos planes de expansión. Quizás con una actualización de su página Web se podría mejorar este aspecto y es por ello que este proyecto también engloba esta opción.

1.4. Clientes

La empresa posee dos tipos de clientes claramente diferenciados: los particulares y los profesionales.

- Cliente particular

Este tipo de clientela puede hacer sus compras de dos formas, acercándose físicamente al establecimiento y escuchando los vinilos in situ o bien vía la página Web.

- Cliente distribuidor

Dentro de los clientes profesionales se puede distinguir de dos tipos:

Promotores y discotecas que quieren realizar eventos de los artistas representados por la empresa.

Actualmente se guardan todos los datos, tanto de promotores como discotecas en archivos Excel. Con la nueva aplicación se pretende que a la hora de realizar un contrato se pueda elegir de una lista al promotor y discoteca que se desee.

Otras tiendas que compren los artículos que tiene en exclusiva la tienda tanto para el territorio nacional como internacional.

1.5. Objetivos

El principal objetivo de este proyecto es conseguir una aplicación que facilite el trabajo en la tienda y consiga ampliar el número de clientes y los beneficios, para lo que será necesario un remodelado completo de la página Web sin perder de vista la seguridad, tanto de los datos de los clientes como de los datos de la empresa.

En un principio las ventas diarias de la tienda se iban anotando en una libreta, luego se pasaron a anotar en archivos EXCEL. Ahora se quiere informatizar todo el proceso, des de la introducción de artículos en la base de datos hasta su venta (tanto al por mayor como al por menor), de manera que sea mucho más fácil controlarlas.

Haciendo esto también se facilitará el saber qué artículo se vende más, qué estilo es el que más demanda tiene. Actualmente no se pueden sacar conclusiones de hacia donde debe dirigirse la tienda ya que no se posee ninguna relación de datos. Con la implantación de la aplicación se podrán obtener unos listados que permitan ajustar costes (si un artista vende pocos discos de su siguiente vinilo se piden menos), maximizar beneficios (cuando un artículo no se vende hay que rebajar su precio) e incluso llegar a saber durante que partes del año hay más ventas para evitar quedarse sin stock.

Otro de los objetivos que se pretende alcanzar con la realización de este proyecto es el llegar a controlar el stock real de la tienda, es decir, saber en todo momento si la cantidad de producto existente en la tienda es el que debería haber. Se necesitan saber las pérdidas por posibles robos, material defectuoso, material dañado, etc.

Desde la creación de la empresa otro de los problemas que han surgido ha sido el de dar a conocer la existencia de la tienda así como de las fiestas organizadas por la misma, ya que la única manera de hacerlo ha sido mediante *flyers*, pósters y el boca a boca; Para solucionar este problema se ha pensado en actualizar la Web, darle más vida, que sea más atractiva para el usuario y crear dentro de la misma un apartado destinado a la información de las fiestas de la empresa. Actualmente la Web presenta un estilo poco intuitivo, a veces complicado y no demasiado usable.

Se puede decir que el objetivo de la empresa es poder crecer aumentando sus beneficios y aprovechando el momento de expansión de este estilo musical pero mejorando la eficiencia en la gestión.

2. ANÁLISIS UML DEL PROYECTO

2.1. Estudio de Viabilidad

2.1.1. Introducción

Los Srs Víctor Peral y Jordi Castilla son los dueños y máximos responsables de la empresa Trivial Musik S.L., empresa dedicada a la venta de música en formato de vinilo y CD y a la representación de artistas a nivel europeo y mundial.

Actualmente se encargan de toda la gestión manualmente, por lo que están interesados en una aplicación informática que les facilite todo su trabajo diario. Esta "automatización" del trabajo aportará grandes ventajas como: tener una base de datos de clientes y distribuidores actualizada, disminuir el tiempo de gestión, facilitar el acceso a sus artículos desde casa (ecommerce)...Sin embargo, como posibles inconvenientes encontramos la familiarización con el nuevo sistema así como su mantenimiento y la fuerte competencia (sobre todo a nivel europeo).

2.1.2. Objeto

a) Descripción de la situación actual

A pesar de la crisis actual las ventas de la tienda y las expectativas de crecimiento de la empresa han aumentado hasta llegar al punto de ser bastante difícil el continuar con los métodos de gestión utilizados hasta el momento.

La empresa además desea expandirse hacia otros mercados y existe la posibilidad de controlar una parte de dicho mercado discográfico así como la representación a nivel nacional e internacional de un gran número de artistas.

b) Perfil cliente-usuario

Habrán 2 tipos de usuarios:

1. Los *administradores* utilizarán las funcionalidades de la aplicación como base de datos para almacenar información. Este tipo de usuario además posee un nivel medio en funcionamiento de aplicaciones de gestión Web, y por tanto, no será necesario darles una formación exhaustiva ya que están familiarizados con este tipo de entornos.

2. Los *clientes* sólo podrán acceder a la Web en sí, es decir, a la tienda online, donde podrán hacer diferentes tipos de búsquedas y acceder a su carrito de la compra. Habrá dos tipos de clientes (cliente normal y cliente distribuidor), la diferencia entre ambos será el precio de los artículos y la cantidad de datos a introducir en el formulario para darse de alta, ya que el cliente distribuidor es un cliente que va a vender nuestros artículos, por lo que será necesario tener su CIF así como el resto de datos de su negocio.

c) Objetivo

El principal objetivo es cambiar y actualizar el sistema de gestión de la empresa, para utilizar de forma más eficiente el tiempo y así poder ampliar los beneficios de la empresa

d) Fuentes de información

1. Aplicaciones Web parecidas: Toolbox Records [<http://www.toolboxrecords.com/>], Juno Records [<http://www.juno.co.uk/>], Pingüins Records [<http://www.pinguins-records.com/>], Switch [<http://www.switchrec.com/>]...
2. Libros para la gestión de bases de datos y tutoriales de MySQL.
3. Libros y manuales de PHP, Javascript i Html para construir la Web.

Ejemplos:

1. Libro: "MASTERING PHPMYADMIN 2.8(PACK PUBLISHING LTD)" ISBN: 1847191606. ISBN-13: 97818471916012.
2. [<http://www.w3c.es/>]. Página oficial de la World Wide Web Consortium, donde se encuentran las normativas para documentos Html.
3. [<http://www.PHP.net>] Página de ayuda sobre el lenguaje de programación PHP.

2.1.3. Sistema a realizar

a) Descripción

El proyecto consistirá en una aplicación Web que permita controlar de forma automática el sistema de gestión de artículos. Esta Web tendrá dos vertientes:

Un gestor de stock que sólo podrán utilizar los administradores y que permitirá introducir, actualizar, eliminar artículos, obtener estadísticas y crear boletines.

Una página Web destinada a los clientes en la cual podrán consultar el stock de artículos así como hacer sus pedidos.

Para controlar el acceso a una u otras partes se hará con diferentes permisos y tendrá una interfaz gráfica intuitiva que permita a cualquier usuario, independientemente de sus conocimientos, hacer búsquedas de artículos así como su compra.

b) Modelo de desarrollo

El modelo será *lineal* porque se entiende que la aplicación se puede dividir en diferentes módulos entre ellos. El hecho de haber trabajado en la tienda durante más de un año ha facilitado la labor a la hora de poder diferenciar las tareas de las que se compone el proyecto.

Por otra parte, toda la información extra necesaria para la creación de la aplicación y de la Web (imágenes, modo de visualizar los datos...) será suministrada por los Srs Víctor Peral y Jordi Castilla.

c) Recursos

Para la realización del proyecto será necesario disponer de un PC para poder ir realizando las pruebas necesarias sobre la página Web y sobre el gestor. Este PC debe tener instalado el siguiente software:

- Sistema Operativo: Microsoft Windows XP Service Pack 2 o posterior.
- Sistema gestor de base de datos: MySQL 6.0 + phpMyAdmin + Apache.
- Programa para la realización de la memoria: Microsoft Office 2003.
- Programa para la realización del "Timing" del proyecto: Microsoft Project 2003.
- Programa para la programación Web (Html, Javascript i PHP): Dreamweaver 8.
- Cliente FTP: FlashFXP.
- Host para poder realizar las pruebas, con opción de FTP, gratuito y compatible con PHP, MySQL i phpMyAdmin.

Una vez realizada la aplicación Web todos los archivos se subirán al servidor de la empresa, i estará ubicada en la página Web: <http://www.trivialmusik.net/>

d) Análisis coste – beneficio

Como beneficios encontramos:

- Un uso más eficiente del tiempo ya que todo el proceso de compra/venta estará informatizado. El sistema nos proporcionará la información necesaria y se podrá dedicar este tiempo a otros planes como el booking (contratación de artistas), eso implica una reducción de los costes y un aumento de los beneficios.
- Un servicio más competitivo a los clientes, ya que el sistema permitirá agilizar todos los trámites necesarios y se podrá atender a los clientes más rápido y más eficientemente, tanto en la tienda física como en la tienda on-line.
- Un aumento del prestigio de la empresa, con la nueva Web será más fácil acceder a un mayor público y hacer crecer a la empresa como marca.
- La comodidad de los clientes de poder realizar sus compras desde casa, el cliente podrá realizar su compra desde cualquier parte del mundo y decidir si recibe el paquete en su casa o pasar a recogerlo por la tienda. Esto también proporcionará un aumento de las ventas y por lo tanto de los beneficios de la empresa.
- La posibilidad de actualizar la Web también desde casa.
- La posibilidad de extraer estadísticas de las ventas realizadas así como de los compradores. Con esto se obtiene un profundo conocimiento de las preferencias de los clientes, por lo que será mucho más sencillo conseguir un aumento de las ventas así como aumentar la fidelidad de los clientes.

Los costes para la empresa de este proyecto serán cero ya que se trata de un proyecto final de carrera.

e) Evaluación de riesgos

El riesgo más elevado es que una vez acabada la aplicación sea demasiado difícil utilizarla. También es importante hacer backups para no perder toda la información de la base de datos además de la seguridad de la misma (estaremos tratando datos confidenciales tales como DNI's, nombres personales...).

Por otra parte es importante que la Web sea compatible con todos los navegadores y que la identificación de los usuarios sea viable mediante un sistema basado en roles.

f) Alternativas

Las alternativas a la aplicación son dos:

- Continuar haciendo el trabajo manualmente, lo que implica un coste 0 pero la pérdida de muchas otras ventajas.
- Encargar la aplicación a una empresa desarrolladora de páginas Web gestoras (por ejemplo, empresa Codalia Research S.L.). Esto conllevaría un coste mínimo de 1.000 €.

2.1.4. Planificación

La ingeniería de software tiene varios modelos o paradigmas de desarrollo en los cuales se puede apoyar para la realización de software, de los cuales podemos destacar a éstos por ser los más utilizados y los más completos:

- Modelo en cascada o Clásico (modelo tradicional)
- Modelo en espiral (modelo evolutivo)
- Modelo de prototipos
- Desarrollo por etapas
- Desarrollo iterativo y creciente o Iterativo e Incremental
- RAD (Rapid Application Development)

Para el desarrollo de este proyecto serán necesarios los siguientes períodos de tiempo para el correcto desarrollo del mismo, por lo que se va a seguir un modelo de desarrollo en cascada pero no riguroso.

Octubre 2008:
Realización del Estudio de Viabilidad.
Análisis de requerimientos.

Noviembre 2008:
Análisis Funcional.
Análisis y diagrama Chen de la Base de Datos.

Diciembre 2008:
Desarrollo y programación de la aplicación Web.

Enero 2009:
Desarrollo y programación de la aplicación Web.

Febrero 2009:
Desarrollo y programación de la aplicación Web.

Marzo 2009:
Fase de pruebas.

Abril 2009:
Fase de pruebas.
Redacción de la memoria del proyecto.

Mayo 2009:
Redacción de la memoria del proyecto.
Desarrollo de la exposición oral.

Junio 2009:
Exposición oral del proyecto.
Entrega de la documentación.

2.1.5. Características del modelo de desarrollo elegido

En Ingeniería de software el desarrollo en cascada, también llamado modelo en cascada, es el enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software, de forma tal que el inicio de cada etapa debe esperar a la finalización de la inmediatamente anterior, debido a esto nuestro proyecto también va a seguir en parte un desarrollo por etapas. El modelo de desarrollo de software por etapas es similar al Modelo de prototipos ya que se muestra al cliente el software en diferentes estados sucesivos de desarrollo, se diferencia en que las especificaciones no son conocidas en detalle al inicio del proyecto y por tanto se van desarrollando simultáneamente con las diferentes versiones del código.

Un ejemplo de una metodología de desarrollo en cascada es:

- 1 Análisis de requisitos: Se analizan las necesidades de los usuarios finales del software para determinar qué objetivos debe cubrir. Es importante señalar que en esta etapa se debe consensuar todo lo que se requiere del sistema, ya que será lo que se realizará en las siguientes etapas, no pudiéndose requerir nuevos resultados a mitad del proceso de elaboración del software.
- 2 Diseño del Sistema: Se descompone y organiza el sistema en elementos que puedan elaborarse por separado, aprovechando las ventajas del desarrollo en equipo.
- 3 Diseño del Programa: Es la fase en donde se realizan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario así como también los análisis necesarios para saber que herramientas usar en la etapa de Codificación.

- 4 Codificación: Es la fase de programación o implementación propiamente dicha. Aquí se implementa el código fuente, haciendo uso de prototipos así como pruebas y ensayos para corregir errores.
- 5 Pruebas: Los elementos, ya programados, se ensamblan para componer el sistema y se comprueba que funciona correctamente antes de ser puesto en explotación.
- 6 Implantación: El software obtenido se pone en producción. Se implantan los niveles software y hardware que componen el proyecto. La implantación es la fase con más duración y con más cambios en el ciclo de elaboración de un proyecto. Es una de las fases finales del proyecto. Durante la explotación del sistema software pueden surgir cambios, bien para corregir errores o bien para introducir mejores. Todo ello se recoge en los Documentos de Cambios.
- 7 Mantenimiento.

De esta forma, cualquier error de diseño detectado en la etapa de prueba conduce necesariamente al rediseño y nueva programación del código afectado, aumentando los costes del desarrollo. La palabra cascada sugiere, mediante la metáfora de la fuerza de la gravedad, el esfuerzo necesario para introducir un cambio en las fases más avanzadas de un proyecto.

2.1.6. Ventajas del modelo de desarrollo elegido

Las ventajas de un desarrollo en cascada son:

1. Se tiene todo bien organizado y no se mezclan las fases.
2. Excelente cuando se tiene un producto estable y se conoce la tecnología.
3. Es un método muy estructurado que funciona bien con gente de poca experiencia.
4. Provee estabilidad en los requerimientos.
5. La planificación se puede hacer anticipadamente.

Las ventajas de un desarrollo por etapas son:

1. Este modelo es útil cuando el cliente conoce los objetivos generales para el software, pero no identifica los requisitos detallados de entrada, procesamiento o salida.
2. También ofrece un mejor enfoque cuando el responsable del desarrollo del software está inseguro de la eficacia de un algoritmo, de la adaptabilidad de un sistema operativo o de la forma que debería tomar la interacción humano-máquina.

2.1.7. Desventajas del modelo de desarrollo elegido

Las desventajas de un desarrollo en cascada son:

1. Poca flexibilidad, no motiva el cambio.
2. Los proyectos en la práctica raramente siguen un flujo secuencial.
3. Siempre es difícil para el cliente mostrar todos los requerimientos explícitamente y con mucha anticipación.
4. Poco apropiado en aplicaciones para la toma de decisiones.
5. Los usuarios tienen una participación limitada.

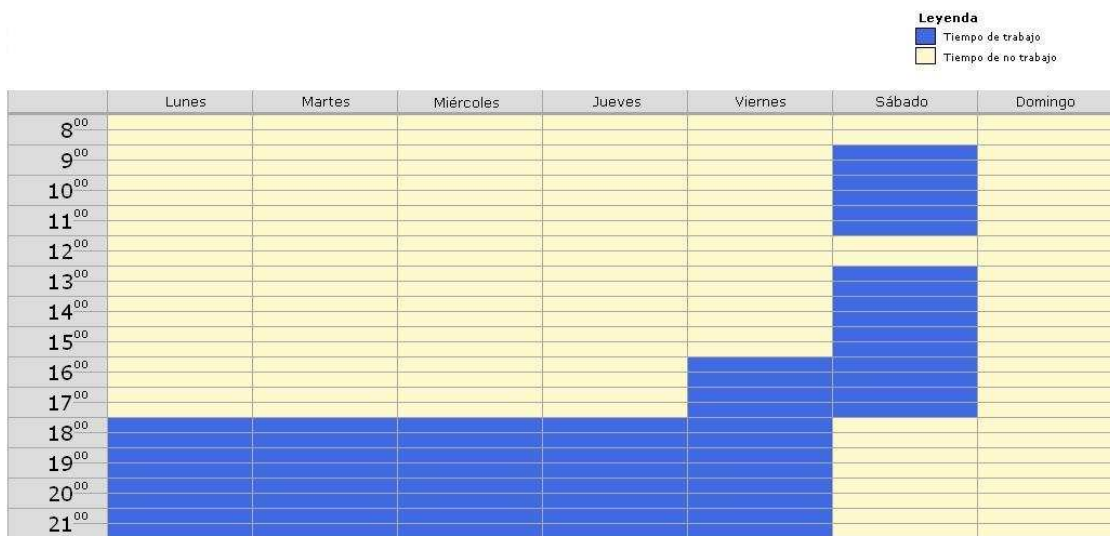
Las desventajas de un desarrollo por etapas son:

- 1 El usuario tiende a crearse unas expectativas cuando ve el prototipo de cara al sistema final. A causa de la intención de crear un prototipo de forma rápida, se suelen desatender aspectos importantes, tales como la calidad y el mantenimiento a largo plazo, lo que obliga en la mayor parte de los casos a reconstruirlo una vez que el prototipo ha cumplido su función. Es frecuente que el usuario se muestre reacio a ello y pida que sobre ese prototipo se construya el sistema final, lo que lo convertiría en un prototipo evolutivo, pero partiendo de un estado poco recomendado.
- 2 En aras de desarrollar rápidamente el prototipo, el desarrollador suele tomar algunas decisiones de implementación poco convenientes (por ejemplo, elegir un lenguaje de programación incorrecto porque proporcione un desarrollo más rápido). Con el paso del tiempo, el desarrollador puede olvidarse de la razón que le llevó a tomar tales decisiones, con lo que se corre el riesgo de que dichas elecciones pasen a formar parte del sistema final.

2.1.8. Planificación temporal del proyecto

En un principio se iba a seguir un fuerte ritmo de trabajo, de tal manera que se pudiese realizar el proyecto rápidamente y para el cual se realizaron las siguientes previsiones:

2.1.8.1. Previsión tiempo de trabajo



Días festivos (días de no trabajo):

- 1 de Noviembre del 2008
- 6 de Diciembre del 2008
- 8 de Diciembre del 2008
- 25 de Diciembre del 2008
- 26 de Diciembre del 2008
- 1 de Enero del 2009
- 6 de Enero del 2009
- 19 de Marzo del 2009
- 11 de Abril del 2009
- 13 de Abril del 2009
- 1 de Mayo del 2009
- 2 de Mayo del 2009
- 8 de Mayo del 2009
- 9 de Mayo del 2009

2.1.8.2. Realidad final del tiempo de trabajo

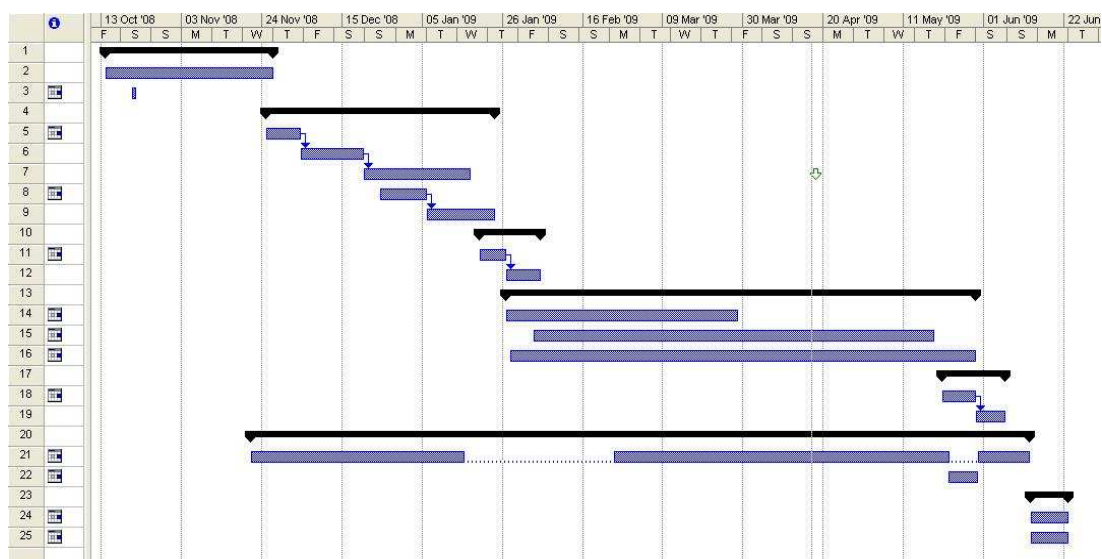
Una vez iniciado el proyecto se vio claramente que no se iban a poder cumplir los períodos previstos.

Esto fue debido a diversas causas como:

- El cambio de trabajo y por lo tanto un cambio de horarios que no permitía dedicar tantas horas a la semana como se hubiese deseado. Se pasó de una estimación de 30 horas semanales a una realidad de 4 o 5 como máximo.
- El echo de comprobar que el trabajo a realizar era mucho mayor de lo esperado en un primer momento.

2.1.8.3. Listado de tareas que representan el desarrollo del proyecto:

		Task Name	Duration	Start	Finish	Predecessors
1		Fase previa	23 days	Mon 13/10/08	Wed 26/11/08	
2		Búsqueda de información	23 days	Mon 13/10/08	Wed 26/11/08	
3		Reunión con el tutor	1 day	Mon 20/10/08	Tue 21/10/08	
4		Análisis	28,5 days	Mon 24/11/08	Fri 23/01/09	
5		Estudio de viabilidad	5 days	Mon 24/11/08	Wed 03/12/08	
6		Análisis de requerimientos	8 days	Wed 03/12/08	Sat 20/12/08	5
7		Análisis funcional	13 days	Sat 20/12/08	Sat 17/01/09	6
8		Análisis de la base de datos	5 days	Wed 24/12/08	Mon 05/01/09	
9		Análisis UML	9 days	Mon 05/01/09	Fri 23/01/09	8
10		Diseño	9 days	Mon 19/01/09	Wed 04/02/09	
11		Diseño de la base de datos	4 days	Mon 19/01/09	Mon 26/01/09	
12		Diseño de la aplicación web	5 days	Mon 26/01/09	Wed 04/02/09	11
13		Codificación	60,5 days	Mon 26/01/09	Fri 29/05/09	
14		Programación XHTML	33 days	Mon 26/01/09	Sat 28/03/09	
15		Programación Javascript	51 days	Mon 02/02/09	Mon 18/05/09	
16		Programación PHP	60 days	Tue 27/01/09	Fri 29/05/09	
17		Pruebas	10 days	Wed 20/05/09	Sat 06/06/09	
18		Pruebas locales	5 days	Wed 20/05/09	Fri 29/05/09	
19		Integración del sistema	5 days	Fri 29/05/09	Sat 06/06/09	18
20		Documentación	#####	Thu 20/11/08	Fri 12/06/09	
21		Redacción de la memoria	76 days	Thu 20/11/08	Fri 12/06/09	
22		Redacción de pruebas y proble...	5 days	Fri 22/05/09	Sat 30/05/09	
23		Presentación	5 days	Sat 13/06/09	Mon 22/06/09	
24		Creación powerpoint	5 days	Sat 13/06/09	Mon 22/06/09	
25		Preparación exposición oral	5 days	Sat 13/06/09	Mon 22/06/09	

2.1.8.4. Diagrama de Gantt:

2.1.9. Conclusiones

Vemos que este proyecto será un gran avance para la empresa Trivial Musik S.L. La creación de una página Web aportará una ampliación de mercado considerable, un aumento de las ventas y dará a conocer a Trivial Musik como marca en sí. Además con el programa de gestión será mucho más sencillo el poder controlar la entrada y salida de artículos de la tienda. Por estas razones, y visto que el coste va a ser casi inexistente este proyecto se puede considerar viable.

2.2. Listado de Requerimientos

Un requerimiento del sistema es una condición o capacidad que tiene que estar presente en un sistema de tal manera que satisfaga una especificación. Por tanto, un requerimiento debe ser conciso y sencillo de entender. Una vez sabido esto y teniendo en cuenta las entrevistas previas realizadas con la empresa, he realizado el documento de análisis de requerimientos recogiendo las especificaciones de lo que el cliente espera que haga la aplicación.

a) Requerimientos Funcionales

Los requerimientos funcionales ofrecen una descripción detallada del comportamiento de la aplicación y de las necesidades del sistema así como soluciones a posibles situaciones adversas o anómalas tales como datos inválidos, errores, fallos del sistema entre otras. Por lo tanto, para poder realizar correctamente el proyecto en cuestión debemos asegurarnos que tenemos claros todos los requerimientos a cumplir. En el siguiente listado veremos que requisitos mínimos debe cumplir nuestra aplicación.

GESTOR DE STOCK:

Evidentes para usuario administrador y ocultos para usuario cliente.

- Se debe crear una base de datos donde almacenaremos todos los datos correspondientes a: clientes, pedidos, productos, distribuidores, enlaces de interés y logs del sistema.
- Debe permitir que el usuario pueda interactuar con las diferentes pantallas pudiendo deshacer.
- Debe pedir clave de acceso al acceder al programa.
- Debe permitir la actualización continua de contenidos, es decir, se debe poder introducir, modificar y eliminar elementos de las bases de datos.
- El diseño debe ser intuitivo.
- Debe validar los campos de los formularios, para campos numéricos como cantidad sólo debe permitir introducir números, para campos con números decimales también debe permitir introducir. , ó `
- Las referencias deben ser guardadas/cargadas en mayúsculas, con 10 caracteres (máximo) no numéricos y 5 numéricos (situados al final).

- Debe permitir hacer ventas al por menor y como distribuidor, donde el valor por defecto será al por menor.
- Cuando se introduzca algún carácter en los campos referencia debe mostrar posibles coincidencias, de forma que pulsando en alguna de las posibles opciones mostradas se cumplimenten todos los campos del formulario.
- Debe permitir guardar los artículos que estén defectuosos, de manera que sea posible extraer estadísticas.
- La aplicación debe guardar las cantidades de artículos vendidas y los precios a los que fueron vendidas.
- Sección para que los usuarios puedan modificar sus datos personales.
- Debe permitir hacer consultas.
- Sección con un formulario para la identificación de los usuarios.
- Sección para poder dar de baja a un administrador.

WEB:

Evidentes para todos los tipos de usuarios.

- Debe permitir que el usuario pueda interactuar con las diferentes pantallas pudiendo deshacer y volver atrás.
- Debe permitir realizar búsquedas de artículos por coincidencia, por sellos y en un futuro se prevé instalar otro buscador por estilo musical.
- Debe contener la posibilidad de que el usuario entre en contacto con la tienda.
- Debe permitir la actualización continua de contenidos.
- Debe soportar conexión a bases de datos.
- Debe ofrecer integridad en las transacciones (datos bancarios, etc.)
- El diseño debe ser intuitivo.
- Debe validar los campos de los formularios antes de su envío.
- Sección para registrar nuevos usuarios, que pida los datos personales i registre el e-mail y una clave para que puedan acceder al site como usuarios registrados.
- Sección con un formulario para la identificación de los usuarios.
- Sección para que los usuarios puedan modificar sus datos personales.
- Sección para que los usuarios puedan darse de baja.

b) Requerimientos no funcionales

Los requerimientos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema. Estos requerimientos se basan en restricciones impuestas por el cliente o bien surgidas por sucesos previstos o imprevistos y que afectan al diseño final. Normalmente son cuantificables. Algunos ejemplos son, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad, mantenimiento, seguridad, portabilidad, estándares, etc.

c) Requerimientos de rendimiento

Especifican restricciones de rendimiento del sistema frente a diferentes situaciones en las que pueda estar inmerso.

Restricciones de diseño:

Las restricciones de diseño son factores presentes en el entorno del cliente que restringen las opciones del diseñador.

Los requerimientos no funcionales de la aplicación son:

- El audio debe estar en formato "mp3", no debe superar los 2.048 Kb, debe estar grabado en mono a 64Kb y no debe superar los tres minutos y medio.
- La imagen debe estar en formato "jpg" o "jpeg" y no debe superar los 8Kb.
- Optimizar al máximo los contenidos para ofrecer tiempos bajos de respuesta y descarga.
- Accesos rápidos a bases de datos.
- Sistema ágil e intuitivo.
- Aviso ante acciones incorrectas.
- En las bases de datos todo se guarda en mayúsculas.
- Control de errores en toda entrada de información con el fin de asegurar su validez, evitando errores como la integridad referencial de la base de datos.
- La página debe ser un sistema multiidioma fácil de adaptar a cualquier tipo de idioma.

2.3. Funcionalidades

En el desarrollo del gestor de stock las funcionalidades sólo van dirigidas a un tipo de usuario, los trabajadores de la tienda. Sin embargo en la aplicación Web las funcionalidades difieren son las mismas independientemente del perfil de usuario.

• Gestor stock:

1. Mantenimiento actualizado de los artículos, es decir, mantenimiento de la base de datos Artículos.
2. Validación de usuario: el usuario deberá identificarse para hacer ciertas acciones concretas como eliminar un artículo de la base de datos, realizar un inventariado...

- Web:

1. Usuario:

Visualización de información general: cualquier usuario de Internet podrá acceder libremente a cualquier parte de la Web, pero no podrá realizar ningún tipo de compra ni apuntarse a ningún tipo de envío de información vía e-mail si no está dado de alta.

Validación de usuario: en el momento que un usuario decida adquirir algún artículo deberá registrarse o identificarse como cliente rellenando un formulario y entrando a formar parte de la base de datos de clientes.

2. Administrador:

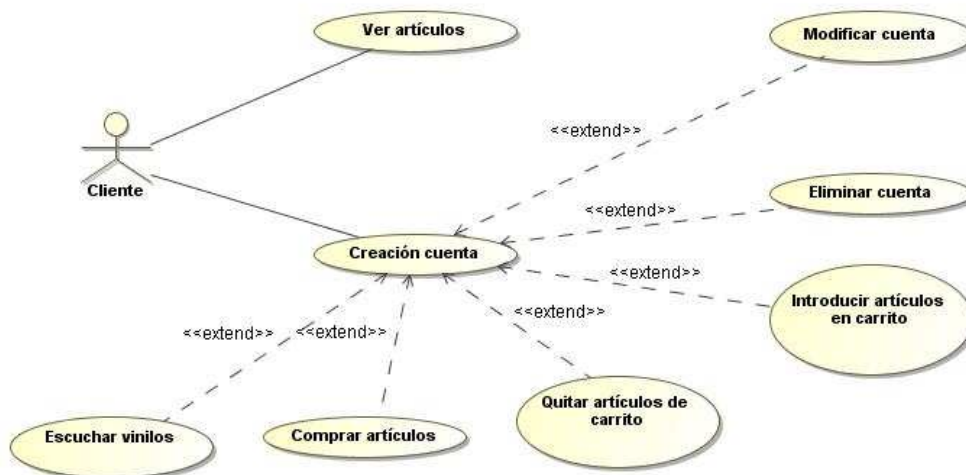
- Validación de usuario: el administrador debe identificarse para acceder a la zona privada de la aplicación.

- Mantenimiento actualizado de los artículos, así como de la información mostrada en la Web.

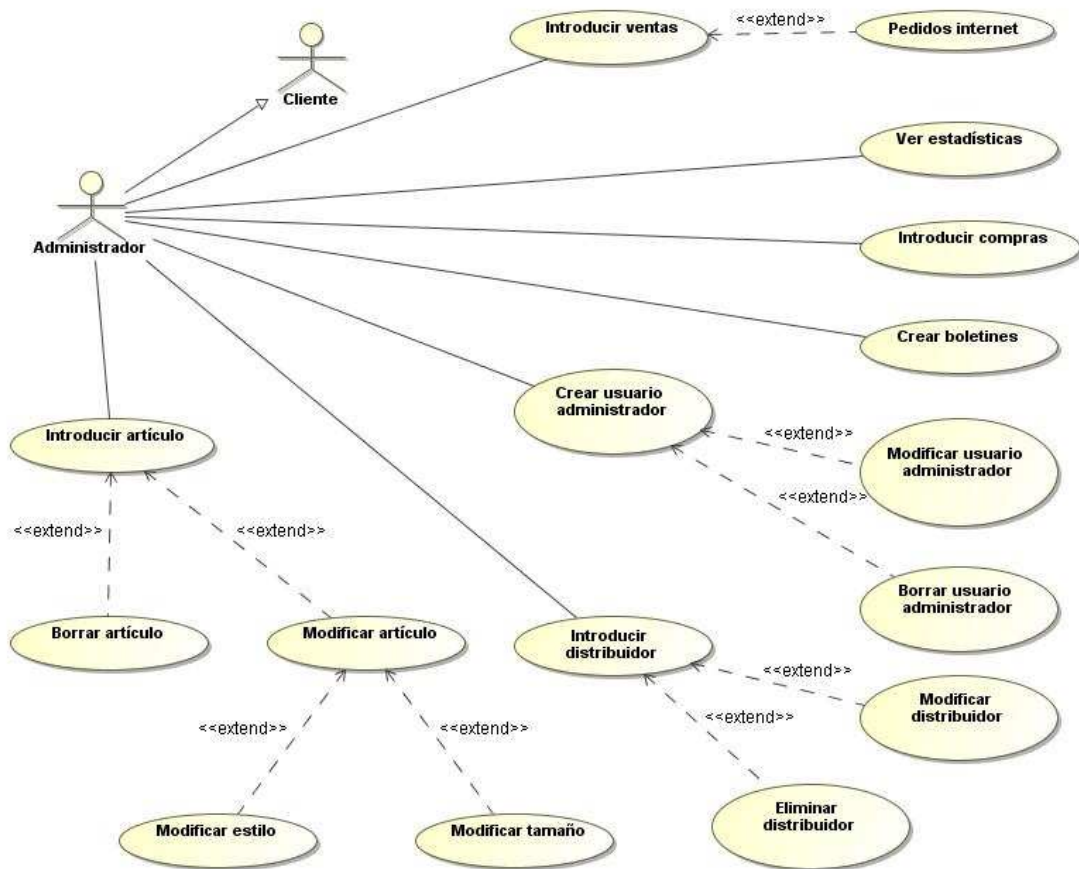
Gestión de la cartera de clientes: El administrador puede añadir, modificar y eliminar los clientes registrados de la base de datos.

2.4. Diagrama casos de uso

2.4.1. Clientes



2.4.2. Administrador



2.5. Casos de uso reales

2.5.1. Gestor de stock

Caso de uso: Identificación de usuario registrado.

Actores: Administrador.

Tipo: Primario.

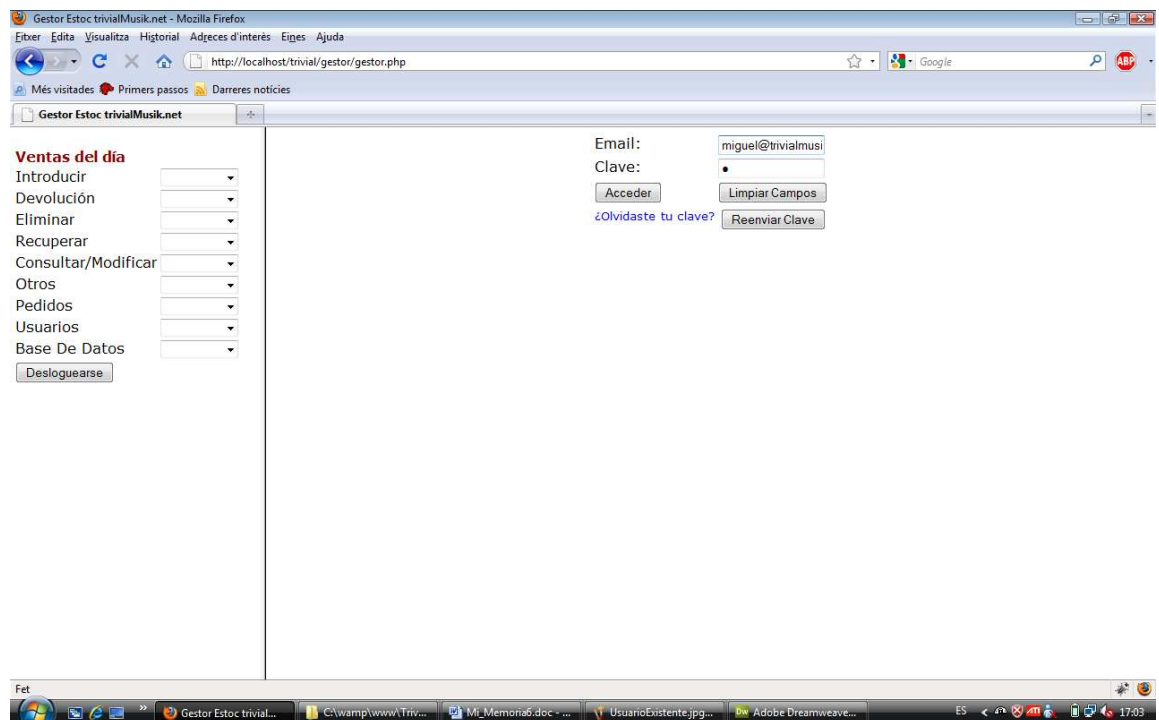
Propósito: Un usuario introduce su e-mail y su password para poder acceder al gestor de stock.

Resumen: El usuario accede a la Web e introduce su e-mail y su clave

Curso normal de los acontecimientos

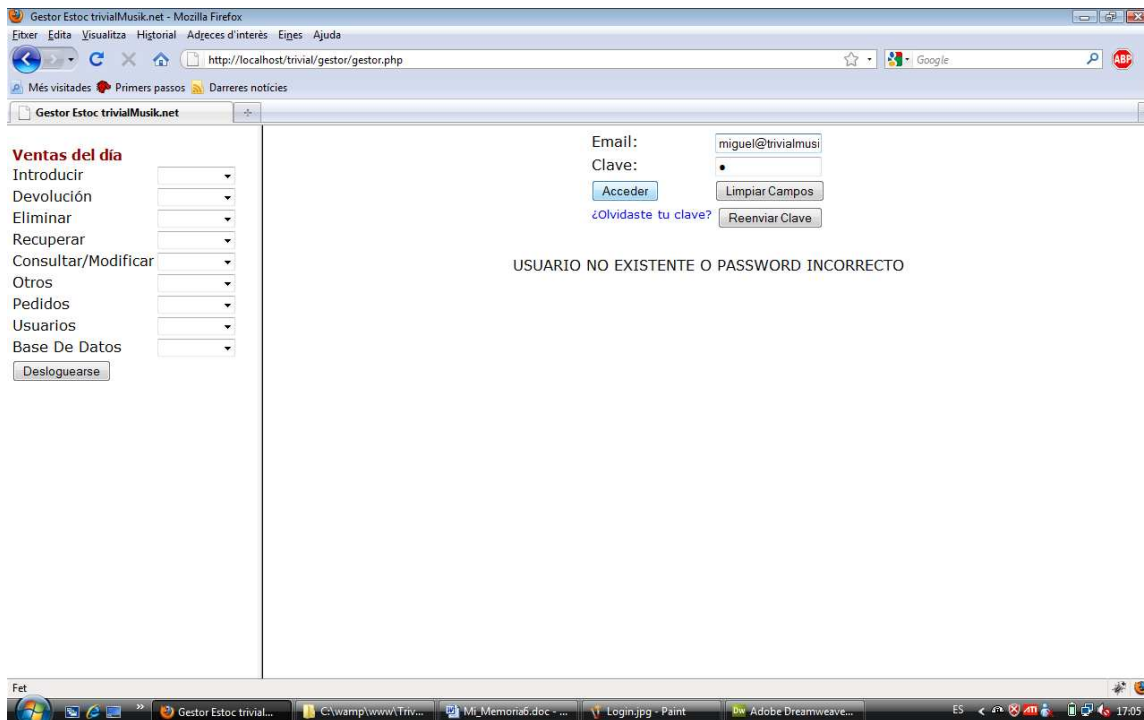
Actores	Sistema
1- El usuario accede a la Web e introduce su e-mail y su clave personal.	2- El sistema recoge los valores de los dos parámetros y verifica que el e-mail exista y que la clave sea correcta. Esta comprobación la realiza haciendo una consulta a la base datos.
	4- Si los datos son correctos se permite al usuario acceder al gestor.

Captura de pantalla de acceso al gestor de stock



Curso alternativo de los acontecimientos

Actores	Sistema
1- El cliente introduce un usuario no introducido en la base de datos o un password incorrecto.	2- El sistema comprueba los datos introducidos y muestra un mensaje de error.
1- El usuario ha olvidado su clave de acceso y pulsa "Reenviar Clave".	2- El sistema envía una clave nueva al e-mail del usuario.

Captura de pantalla de acceso al gestor de stock, acceso incorrecto

Caso de uso: Desloguearse de la Web un usuario registrados.

Actores: Administrador.

Tipo: Primario.

Propósito: El usuario no sigue trabajando con el gestor.

Resumen: El usuario pulsa en el menú "Desloguearse".

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Desloguearse".	2- El sistema redirecciona al cliente a una página Web dinámica indicando que la operación se ha llevado a cabo correctamente y vuelve a pedir e-mail y clave.

Caso de uso: Hacer una venta

Actores: Administrador.

Tipo: Primario.

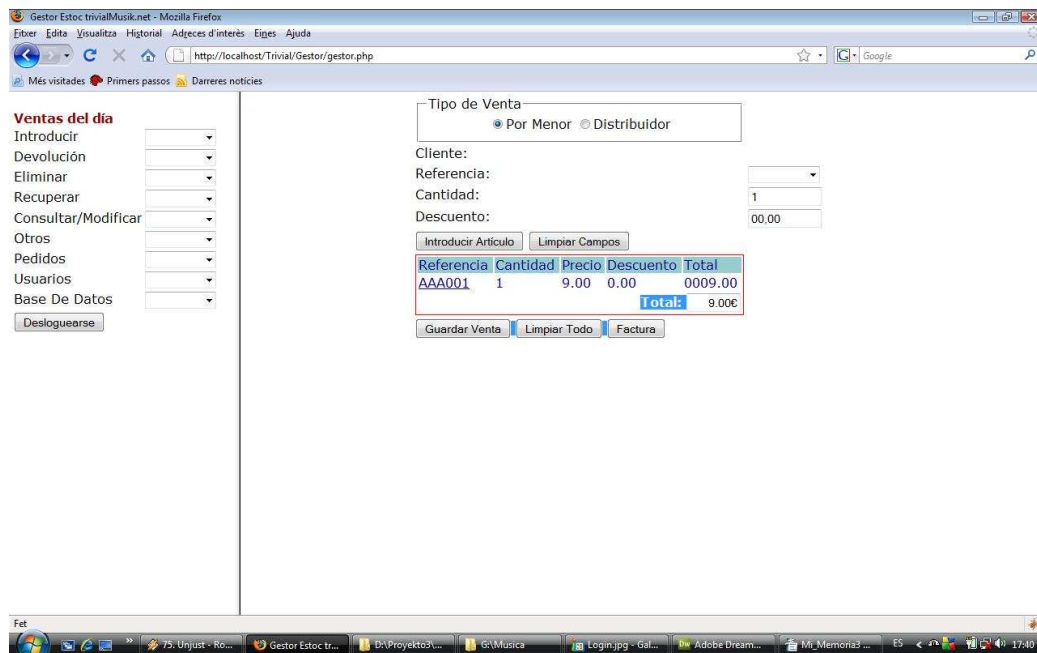
Propósito: Registrar una venta en la base datos, ya sea en la tienda o una venta on-line.

Resumen: El usuario accede al gestor de stock y realiza una venta (pantalla por defecto al acceder al gestor y una vez logueado el usuario)

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario accede al gestor stock.	2- El sistema muestra una página Web dinámica con un formulario.
3- El usuario selecciona el tipo de venta por defecto "Por Menor", también selecciona la referencia del artículo (lista desplegable), la cantidad de artículos y si poseen descuento.	5- El sistema hace unas comprobaciones e introduce el artículo en la lista de la venta actual.
4- El usuario pulsa "Introducir artículo".	5- El sistema introduce el artículo en la lista actual y muestra los datos de la venta, hasta el momento.
6- El usuario pulsa "Guardar Venta".	6- El sistema introduce los datos en la base datos, queda registrada la venta.

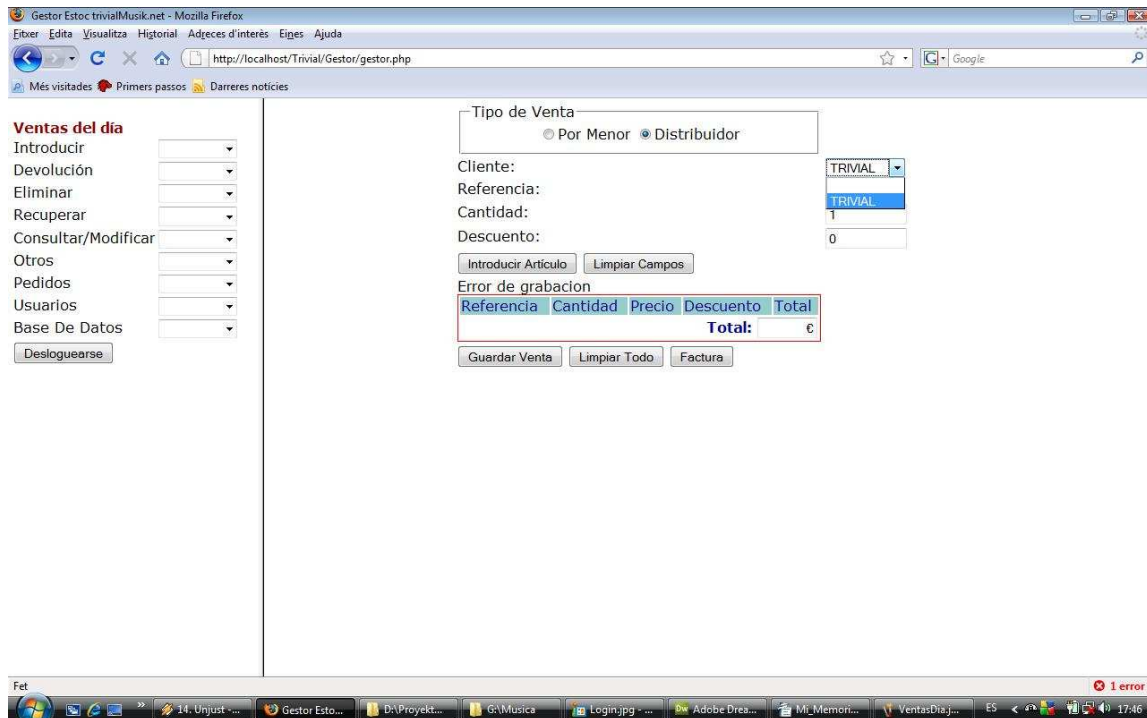
Captura de pantalla de venta



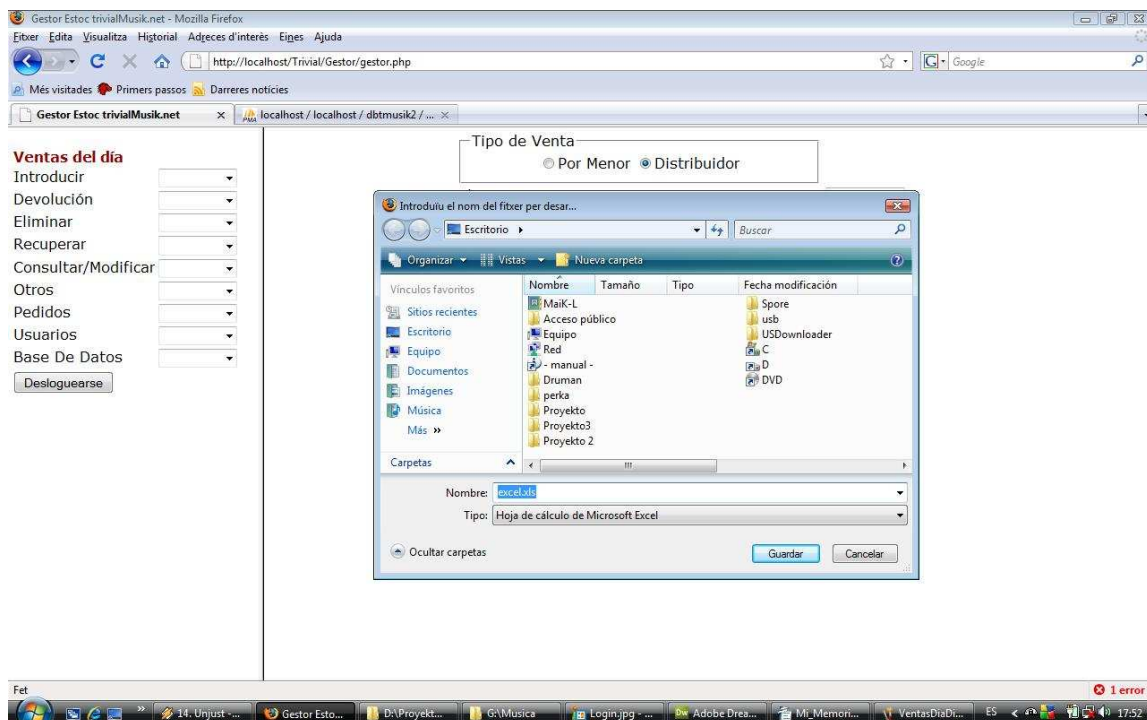
Cursos alternativos de los acontecimientos

Actores	Sistema
1- El usuario selecciona tipo de venta "Distribuidor".	2- El sistema actualiza dinámicamente la página y muestra desplegable en el que seleccionar a que cliente se le va a hacer la venta.
1- El usuario antes de finalizar la venta pulsa el botón "Factura".	2- El sistema responde preguntando al cliente donde quiere guardar el archivo.
3- El usuario selecciona una carpeta.	4- El sistema crea el archivo en formato .xls
1- El usuario selecciona más artículos de los que se dispone según el sistema.	2- El sistema devuelve mensaje de "Error de grabación ", el artículo no se introduce en la lista
1- El usuario introduce cantidad cero, deja la casilla en blanco o no selecciona ninguna referencia.	2- El sistema devuelve pop up indicando error.
1- El usuario desea quitar un artículo de la lista y pulsa sobre la referencia del mismo en la lista de venta.	2- El sistema elimina el artículo de la lista.

Captura de pantalla venta a cliente Distribuidor



Captura de pantalla de creación de factura



Ejemplo de factura

FacturaEjemplo - OpenOffice.org Calc

Archivo Editar Ver Insertar Formato Herramientas Datos Ventana Ayuda

Arial 10

E12

	A	B	C	D	E	F	G
1							
2	Fecha: 06/06/2009				Cliente: 0000000000		
3							
4	CONCEPTO	PREU	DESCOMPTI	#	SUMA		
5	AAA001	01/09/00	0.00	1	0009.00		
6	AAA002	01/09/00	0.00	1	0009.00		
7	BBB002	01/09/00	01/05/00	1	0004.00		
8	MGB001	01/09/00	01/03/50	1	0005.50		
9			Total	4	27.50		
10							
11							
12							
13							
14							
15							

Introducir Artículo Limpiar Campos

Registro grabado correctamente

Referencia	Cantidad	Precio	Descuento	Total
AAA001	1	9.00	0.00	0009.00
AAA002	1	9.00	0.00	0009.00
BBB002	1	9.00	5.00	0004.00
MGB001	1	9.00	3.50	0005.50
Total:				27.50€

Guardar Venta Limpiar Todo Factura

Captura de pantalla de error de grabación

Gestor Estoc trivialMusik.net - Mozilla Firefox

Editar Visualiza Historial Adreces d'interès Eines Ajuda

http://localhost/Trivial/Gestor/gestor.php

Més visites Primers passos Darreres notícies

Gestor Estoc trivialMusik.net localhost / localhost / dbtmusik2 / ...

Ventas del día

Introducir
Devolución
Eliminar
Recuperar
Consultar/Modificar
Otros
Pedidos
Usuarios
Base De Datos
Desloguearse

Tipo de Venta
☒ Por Menor ☐ Distribuidor

Cliente:
Referencia:
Cantidad: 1
Descuento: 0

Introducir Artículo Limpiar Campos

Error de grabacion

Referencia	Cantidad	Precio	Descuento	Total
AAA001	1	9.00	0.00	0009.00
MGB001	5	9.00	0.00	0045.00
Total:				54.00€

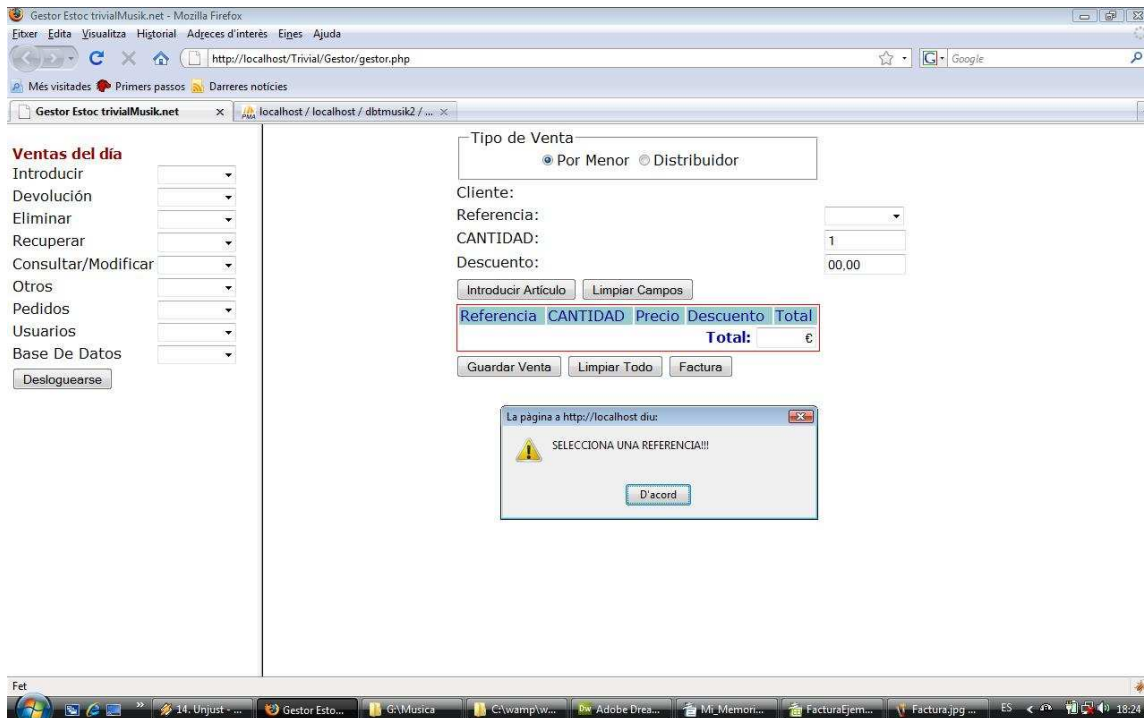
Guardar Venta Limpiar Todo Factura

Fet

14: Unjust - Ro... Gestor Estoc tr... G:\Musica Login.jpg - Gal... Adobe Dream... Mi_Memoria3... VentaFactura.j... ES 17:59

2 errors

Captura de pantalla error al introducir datos venta



Caso de uso: Crear nuevo usuario.

Actores: Administrador.

Tipo: Primario.

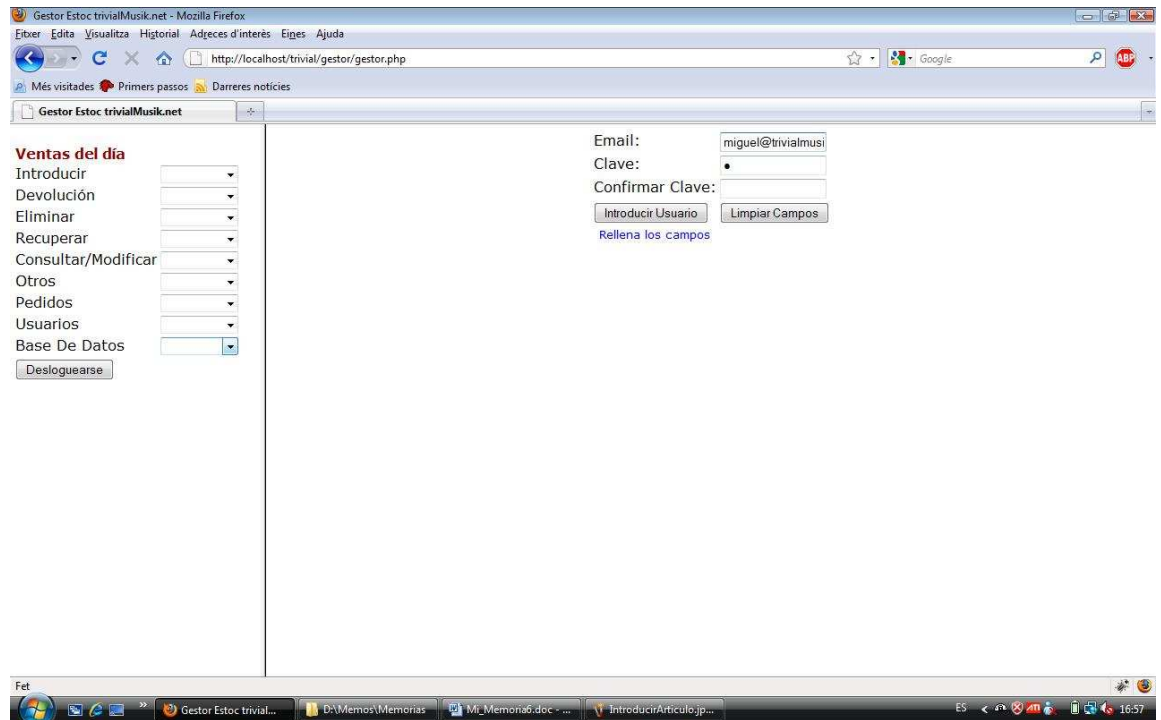
Propósito: Dar permisos a alguien para que pueda acceder al gestor.

Resumen: Un usuario logueado puede dar permisos a alguien para que pueda acceder al gestor.

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Usuario" -> "Nuevo Usuario"	2- El sistema devuelve una página Web dinámica con un formulario en el que se pide e-mail, clave y confirmación de clave.
3- El usuario introduce los datos y pulsa "Introducir Usuario"	4- El sistema hace sus comprobaciones y muestra el mensaje "USUARIO Introducido CORRECTAMENTE"

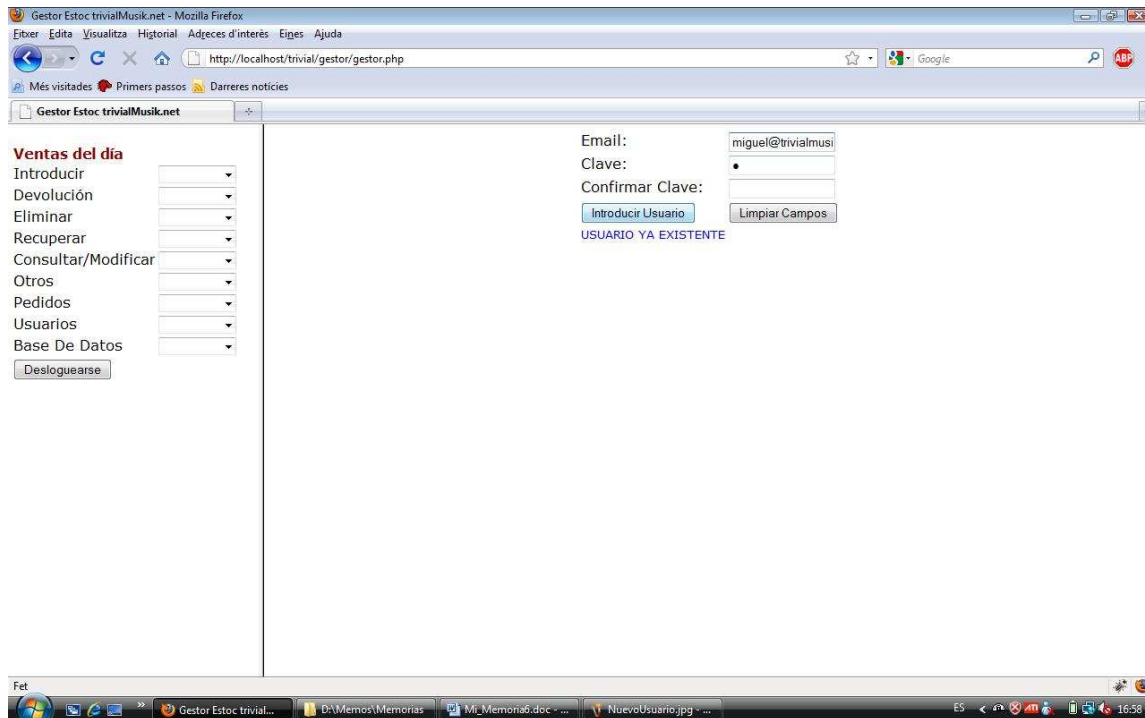
Captura de pantalla de creación de nuevo usuario



Curso alternativo de los acontecimientos

Actores	Sistema
1- El cliente introduce una clave y una confirmación de clave diferentes	2- El sistema comprueba los datos introducidos y redirecciona a una Web dinámica avisando del error cometido y pidiendo los datos de nuevo.
1- El cliente introduce un e-mail ya existente en la base de datos	2- El sistema comprueba los datos introducidos y muestra mensaje de error.

Captura de pantalla creación incorrecta de usuario



Caso de uso: Crear backup de la base de datos.

Actores: Administrador.

Tipo: Primario.

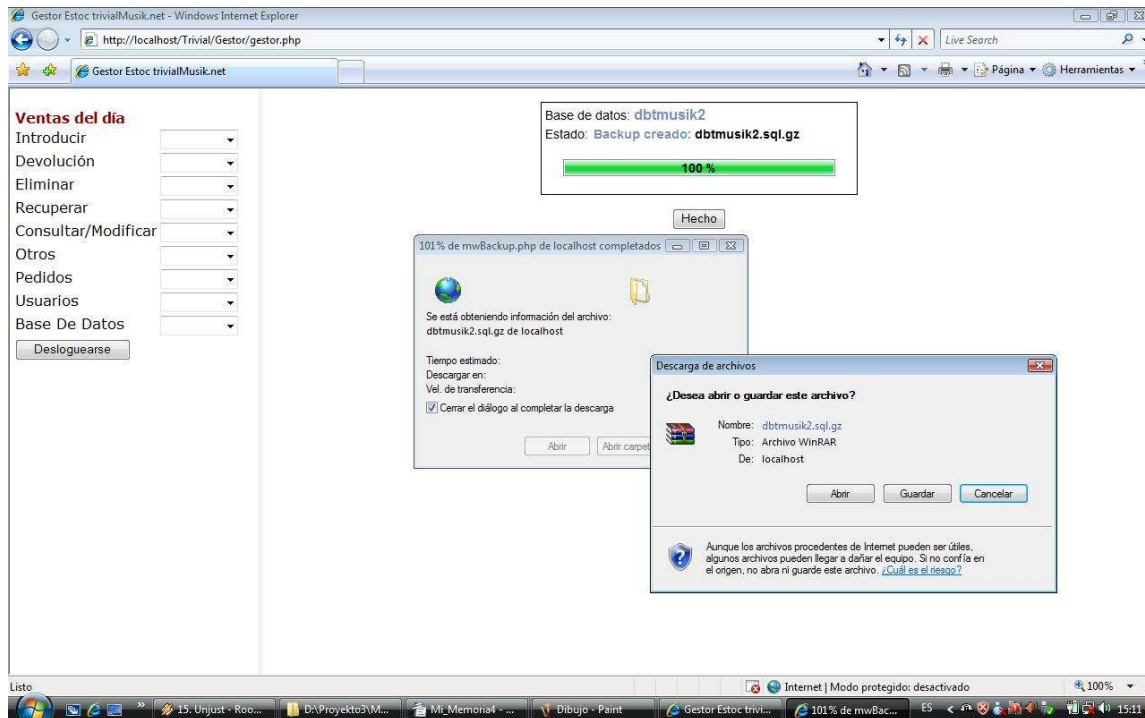
Propósito: Crear una copia de seguridad de la base de datos.

Resumen: Crear una copia de seguridad de la base de datos

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Bases de datos" > "Hacer Backup"	2- El sistema redirecciona al cliente a una página Web dinámica preguntando si desea hacer el backup comprimido o no.
3- El usuario elige una opción y pulsa proceder	4- El sistema hace el backup y pregunta donde se desea guardar el archivo de seguridad.
5- El usuario pulsa "Hecho"	6- El sistema redirige al usuario a la pantalla de "Ventas del Día"

Captura de pantalla al crear el backup de la aplicación



Caso de uso: Cargar backup de la base de datos.

Actores: Administrador.

Tipo: Primario.

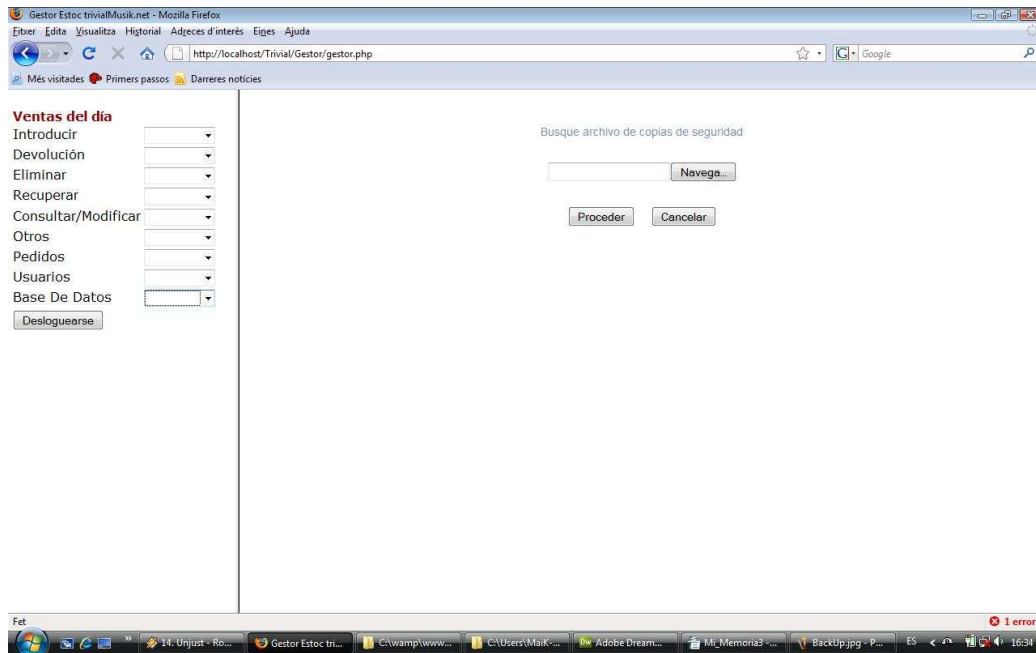
Propósito: Cargar una copia de seguridad de la base de datos.

Resumen: Cargar una copia de seguridad de la base de datos

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Bases de datos" > "Recuperar"	2- El sistema redirecciona al cliente a una página Web pidiendo el archivo que quiere cargar.
3- El usuario selecciona el archivo a cargar y pulsa "Proceder"	4- El sistema hace las comprobaciones pertinentes y carga los datos en la base de datos. Muestra mensaje que el proceso ha finalizado correctamente.

Captura de pantalla cargar archivo de recuperación de base de datos



Curso alternativo de los acontecimientos

Actores	Sistema
1- El cliente no selecciona ningún archivo	2- El sistema muestra mensaje de: "No ha seleccionado ningún archivo para RESTORE."
3- El usuario pulsa "Cancelar"	4- El sistema redirige al usuario a la pantalla de "Ventas del Día"

Caso de uso: Introducir compras, artículos, sellos, distribuidores o fabricantes

Actores: Administrador.

Tipo: Primario.

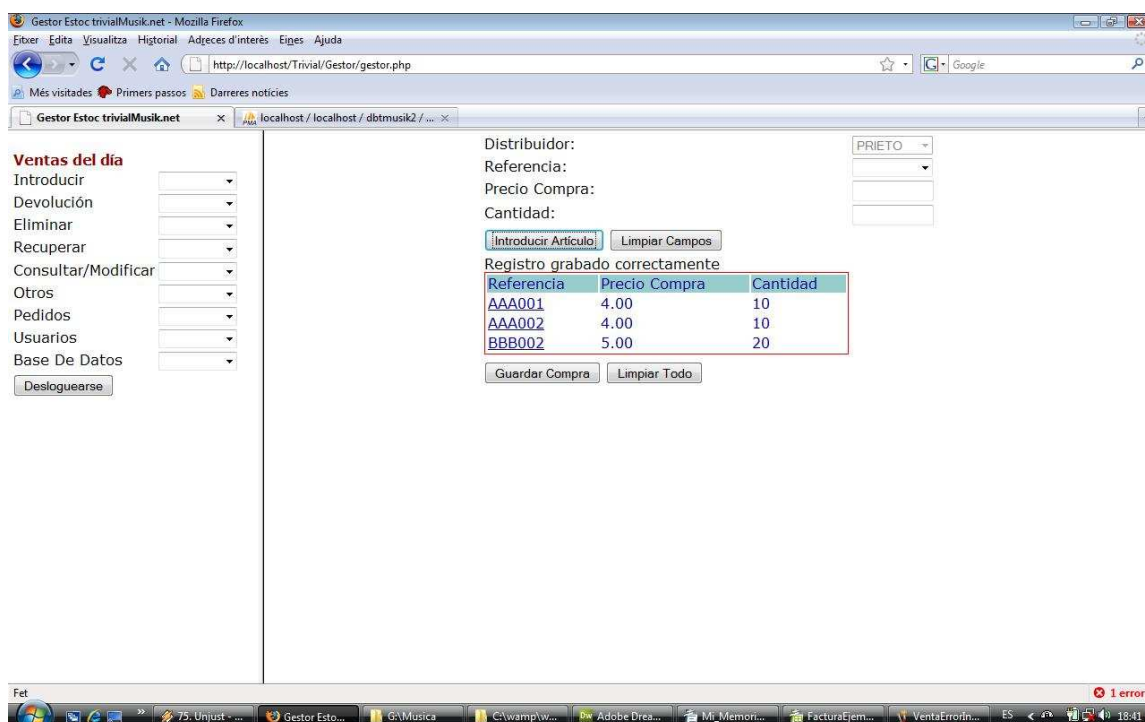
Propósito: Introducir compras, artículos, sellos, distribuidores o fabricantes en la base datos.

Resumen: Introducir compras, artículos, sellos, distribuidores o fabricantes en la base datos.

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Introducir" > La opción deseada.	2- El sistema redirecciona al cliente a una página Web mostrando un formulario u otro dependiendo de lo que se quiera introducir en la base de datos.
3- El usuario introduce los datos pedidos.	4- El sistema hace las comprobaciones pertinentes e introduce los datos en la base de datos

Captura de pantalla introducir compra



Captura de pantalla introducir artículo

Referencia:

Número Referencia:

Precio minorista:

Precio mayorista:

Artista:

Disco:

Tamaño:

Estilo:

Tipo:

Descripción:

tracklist:

Imagen: Navega...

Sample: Navega...

Captura de pantalla introducir sello

Nombre del Sello:

Referencia del Sello:

Fabricante:

Dueño(s):

Email1:

Email2:

Web1:

Web2:

Telf1:

Telf2:

Registro grabado correctamente

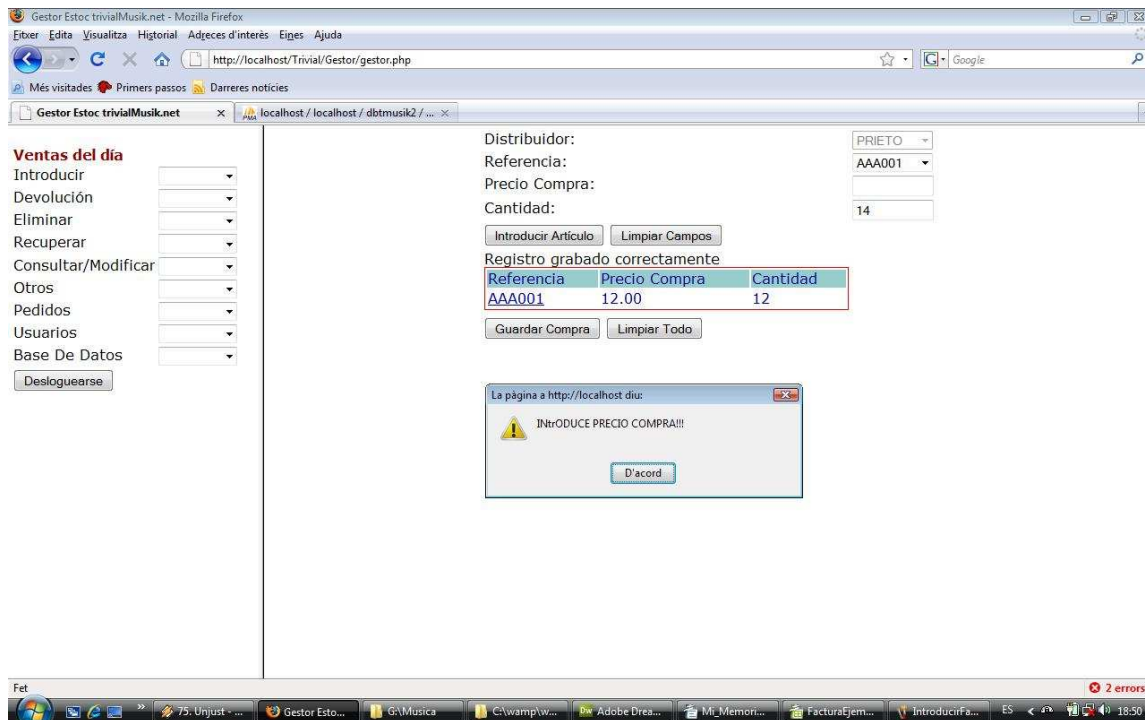
Referencia	Nombre Sello
GTEK	GANJATEK

Captura de pantalla introducir distribuidor

Curso alternativo de los acontecimientos

Actores	Sistema
1- El usuario introduce mal un dato o lo deja en blanco y es un campo obligatorio.	2- El sistema devuelve mensaje de error.
1- El usuario intenta introducir un artículo, sello, referencia o fabricante ya existente en la base de datos.	2- El sistema devuelve mensaje de error.
1- El usuario desea quitar un artículo de la compra a introducir o un sello del listado a introducir.	2- El usuario pulsa sobre la referencia o sobre el sello y el sistema lo elimina de la lista.

Captura de pantalla de dato introducido incorrectamente



Caso de uso: Devolución de artículos a un distribuidor

Actores: Administrador.

Tipo: Primario.

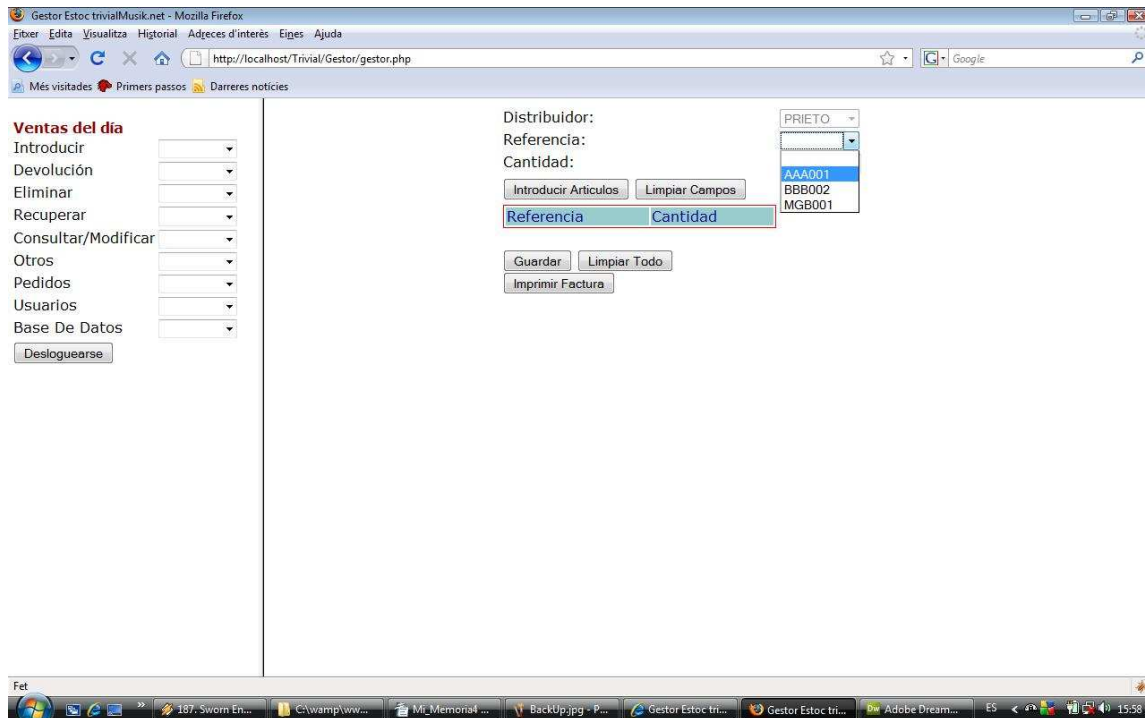
Propósito: Devolver artículos a uno de los distribuidores por algún error de envío o defecto.

Resumen: Devolver artículos a uno de los distribuidores.

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Devolución" > "Devol.Distr."	2- El sistema redirecciona al cliente a una página Web mostrando un formulario.
3- El usuario selecciona un distribuidor, una referencia y la cantidad a devolver.	4- El sistema hace las comprobaciones pertinentes e introduce los datos en la base de datos

Captura de pantalla Devolución Distribuidor



Curso alternativo de los acontecimientos

Actores	Sistema
1- El usuario no introduce ninguna cantidad y pulsa "Introducir artículos"	2- El sistema comprueba los datos introducidos y muestra un mensaje de error.
1- El usuario desea eliminar uno de los artículos introducidos en la lista de artículos a devolver y pulsa sobre la referencia que desea eliminar.	2- El sistema elimina el artículo de la lista.

Caso de uso: Cambio de artículo

Actores: Administrador.

Tipo: Primario.

Propósito: Cambiar un artículo a un cliente minorista.

Resumen: Cambiar un artículos a un cliente porque estaba defectuoso o alguna otra razón.

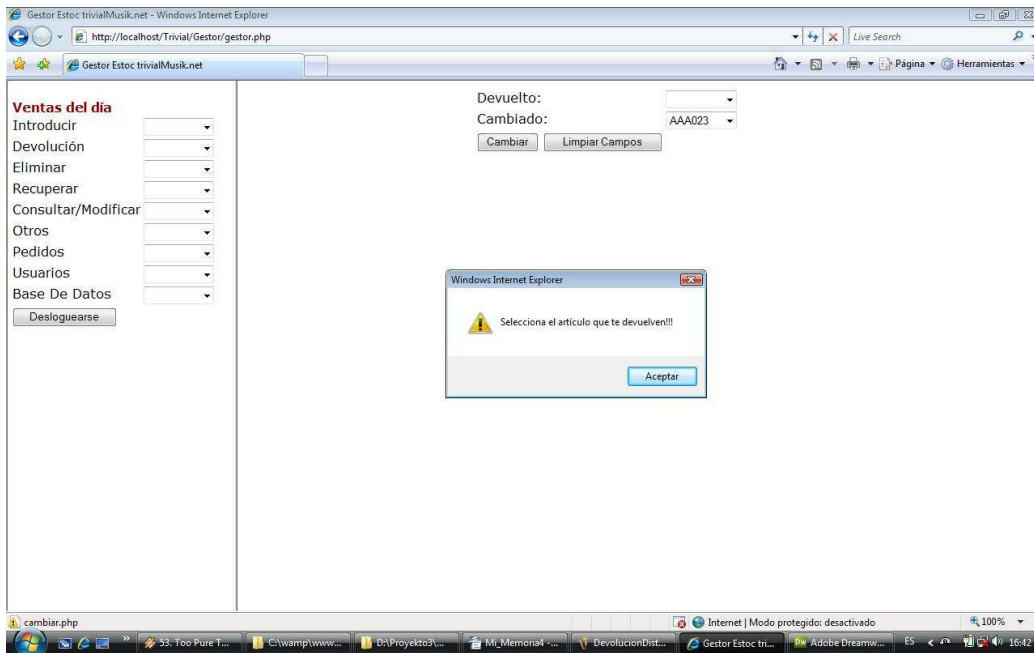
Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Devolución" > "Cambio"	2- El sistema redirecciona al cliente a una página Web mostrando un formulario.
3- El usuario selecciona el artículo devuelto así como por el que se cambia.	4- El sistema hace las comprobaciones pertinentes e introduce los datos en la base de datos. 5- El sistema redirige al usuario al formulario de cambio de artículo.

Curso alternativo de los acontecimientos

Actores	Sistema
1- El usuario no selecciona uno de los campos posibles.	2- El sistema muestra pop-up mostrando el error.

Captura de pantalla error al realizar cambio artículo



Caso de uso: Eliminar artículo, sello, distribuidor o fabricante.

Actores: Administrador.

Tipo: Primario.

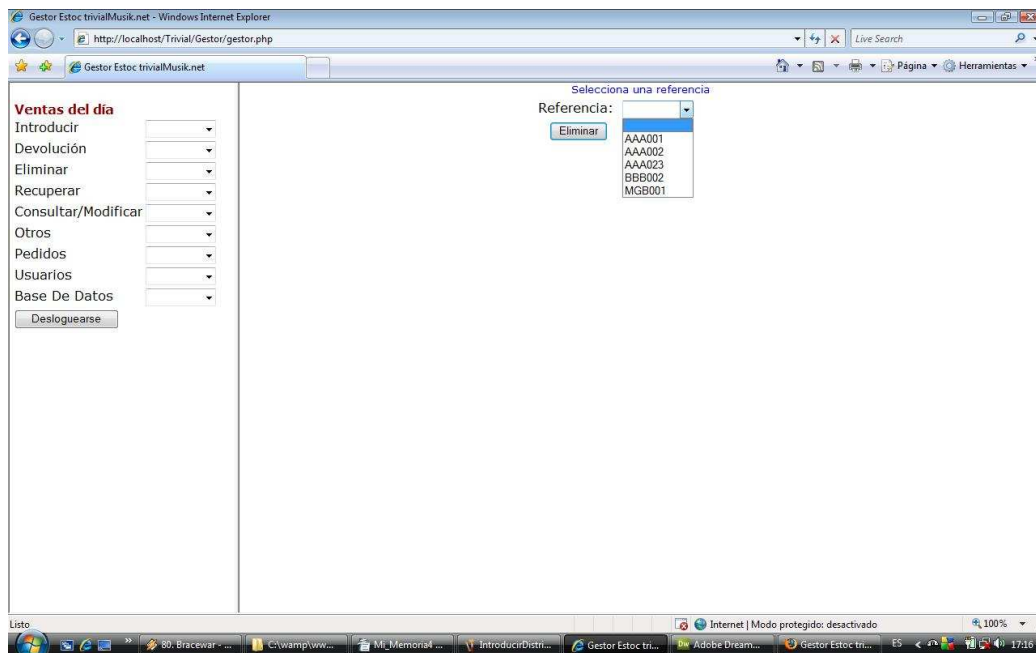
Propósito: Eliminar artículo, sello, distribuidor o fabricante.

Resumen: Eliminar artículo, sello, distribuidor o fabricante.

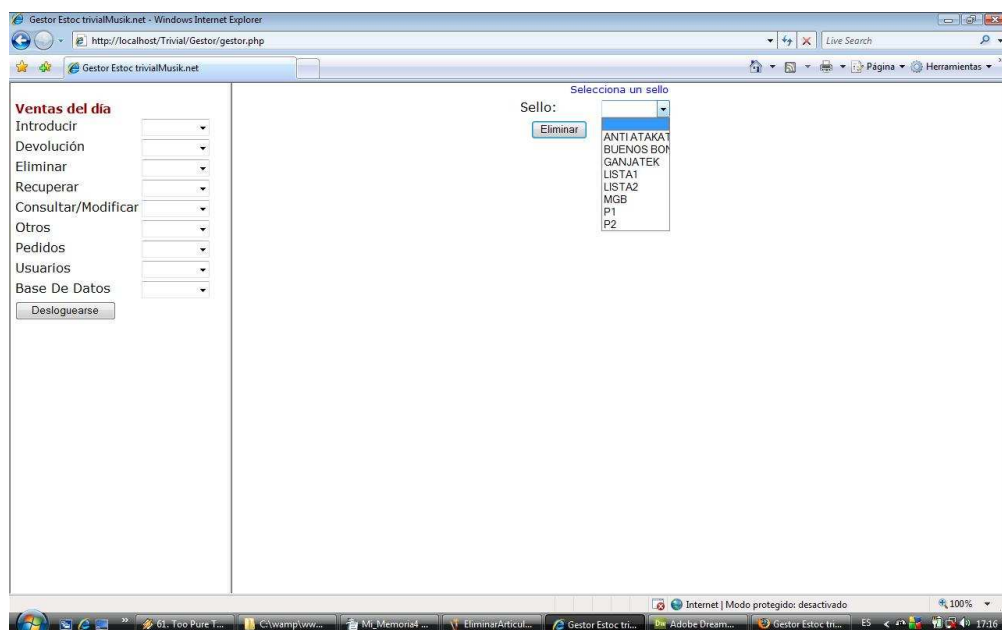
Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Eliminar" > la opción deseada	2- El sistema redirecciona al cliente a una página Web mostrando un formulario.
3- El usuario selecciona una de las opciones del menú desplegable mostrado.	4- El sistema hace las comprobaciones pertinentes y elimina la opción escogida de la base de datos.

Captura de pantalla eliminar artículo

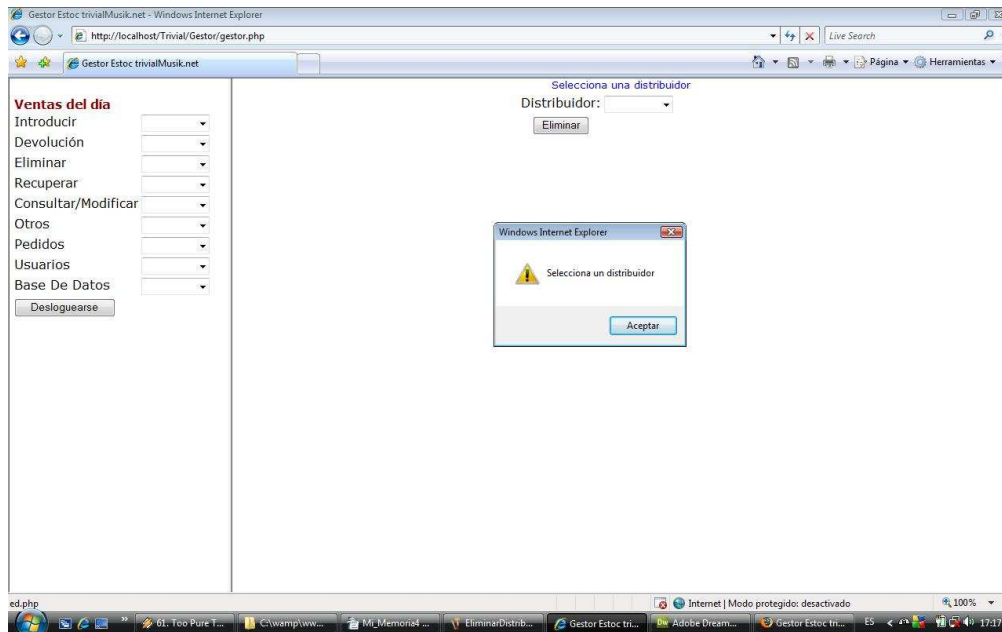


Captura de pantalla eliminar sello



Curso alternativo de los acontecimientos

Actores	Sistema
1- El usuario no selecciona ninguna opción.	2- El sistema muestra pop-up mostrando el error.

Captura de pantalla error al eliminar

Caso de uso: Recuperar artículo, sello, distribuidor o fabricante.

Actores: Administrador.

Tipo: Primario.

Propósito: Recuperar artículo, sello, distribuidor o fabricante.

Resumen: Recuperar artículo, sello, distribuidor o fabricante.

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Recuperar" > la opción deseada	2- El sistema redirecciona al cliente a una página Web mostrando un formulario que contiene un desplegable con las posibles opciones a escoger
3- El usuario selecciona una de las opciones del menú desplegable mostrado.	4- El sistema hace las comprobaciones pertinentes y elimina la opción escogida de la base de datos.

Curso alternativo de los acontecimientos

Actores	Sistema
1- El usuario no selecciona ninguna opción.	2- El sistema muestra pop-up mostrando el error.

Caso de uso: Consultar/Modificar artículo, sello, distribuidor o fabricante.

Actores: Administrador.

Tipo: Primario.

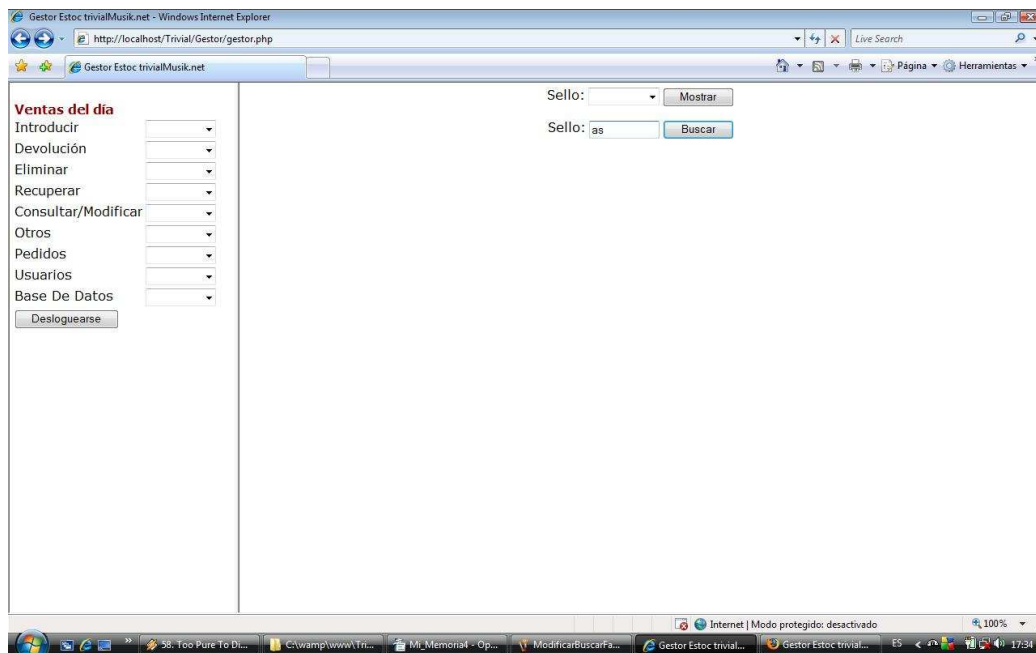
Propósito: Consultar/Modificar, sello, distribuidor o fabricante.

Resumen: Consultar/Modificar, sello, distribuidor o fabricante.

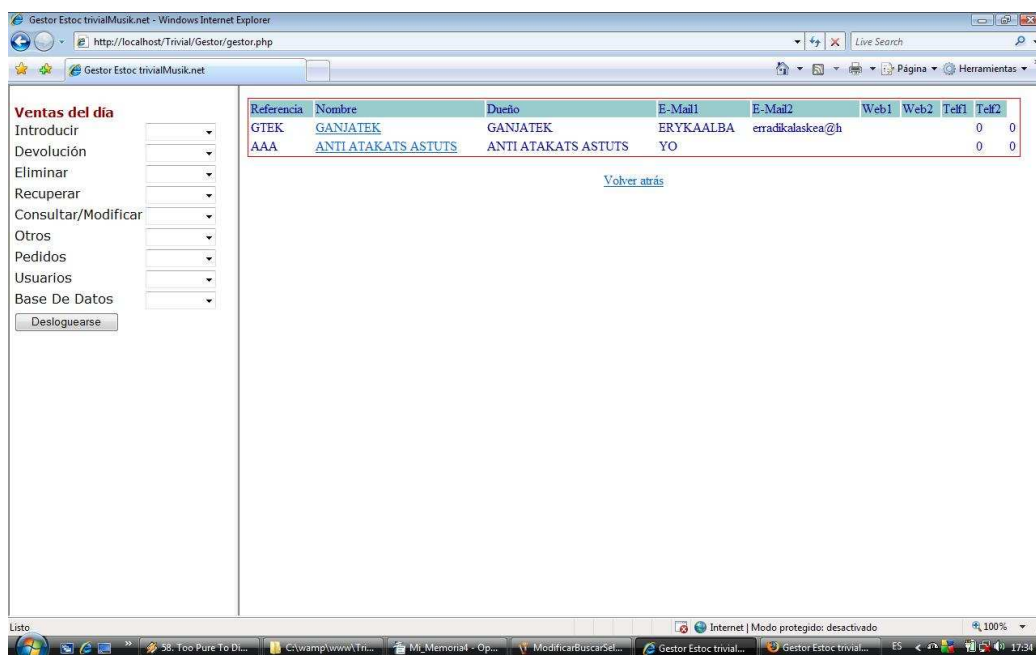
Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Consultar/Modificar" > la opción deseada.	2- El sistema redirecciona al cliente a una página Web mostrando un formulario que contiene un desplegable con las posibles opciones a escoger y una casilla de búsqueda (búsqueda por coincidencia) en la que se puede introducir texto.
3- El usuario introduce un texto a buscar.	4- El sistema muestra un listado con las posibles opciones.
5- El usuario pulsa en una de las opciones.	6- El sistema muestra los datos de la opción escogida.
7- El usuario modifica algún dato y pulsa "Guardar".	8- El sistema hace las comprobaciones necesarias y guarda los cambios en la base de datos.

Captura de pantalla Consultar/Modificar sello



Captura de pantalla listado sellos



Captura de pantalla mostrar datos sello

Nombre del Sello: GANJATEK

Referencia del Sello: GTEK

Fabricante:

Dueño(s): ERYKAALBA

Email1: erradikalaske

Email2:

Web1:

Web2:

Tel1: 0

Tel2: 0

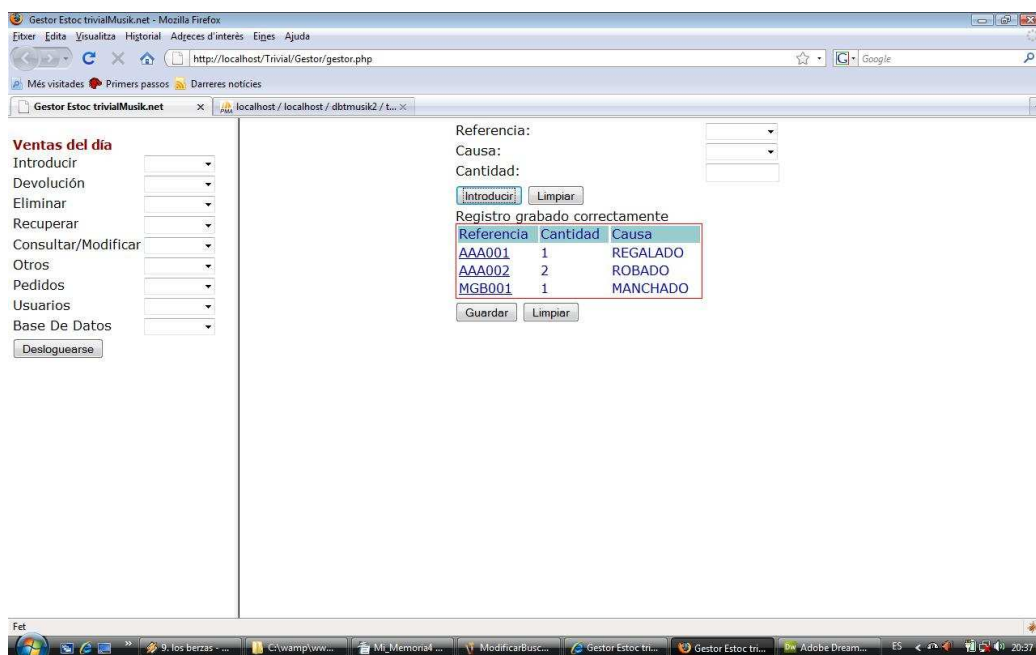
Curso alternativo de los acontecimientos

Actores	Sistema
1- El usuario no selecciona ninguna opción.	2- El sistema muestra pop-up mostrando el error.
1- El usuario introduce texto de búsqueda.	2- El sistema no encuentra ninguna coincidencia y muestra lista en blanco.
3- El usuario pulsa "Volver Atrás".	4- El sistema le redirige a la página de selección/búsqueda.

Caso de uso: Marcar artículo como defectuoso.**Actores:** Administrador.**Tipo:** Primario.**Propósito:** Marcar artículo como defectuoso.**Resumen:** Marcar artículo como defectuoso.

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Otros" > "Defectuosos".	2- El sistema redirecciona al cliente a una página Web mostrando un formulario.
3- El usuario rellena el formulario y pulsa "Introducir"	4- El sistema introduce el artículo en la
5-El usuario pulsa "Guardar"	4- El sistema introduce el/los artículos
	6- El sistema guarda los artículos como defectuosos.

Captura de pantalla marcar artículo como defectuoso**Curso alternativo de los acontecimientos**

Actores	Sistema
1- El usuario deja de rellenar alguna casilla del formulario.	2- El sistema muestra pop-up mostrando el error.
1- El usuario desea quitar algún elemento de la lista y pulsa sobre la referencia.	2- El sistema elimina el artículo de la lista.

Caso de uso: Obtener listado.

Actores: Administrador.

Tipo: Primario.

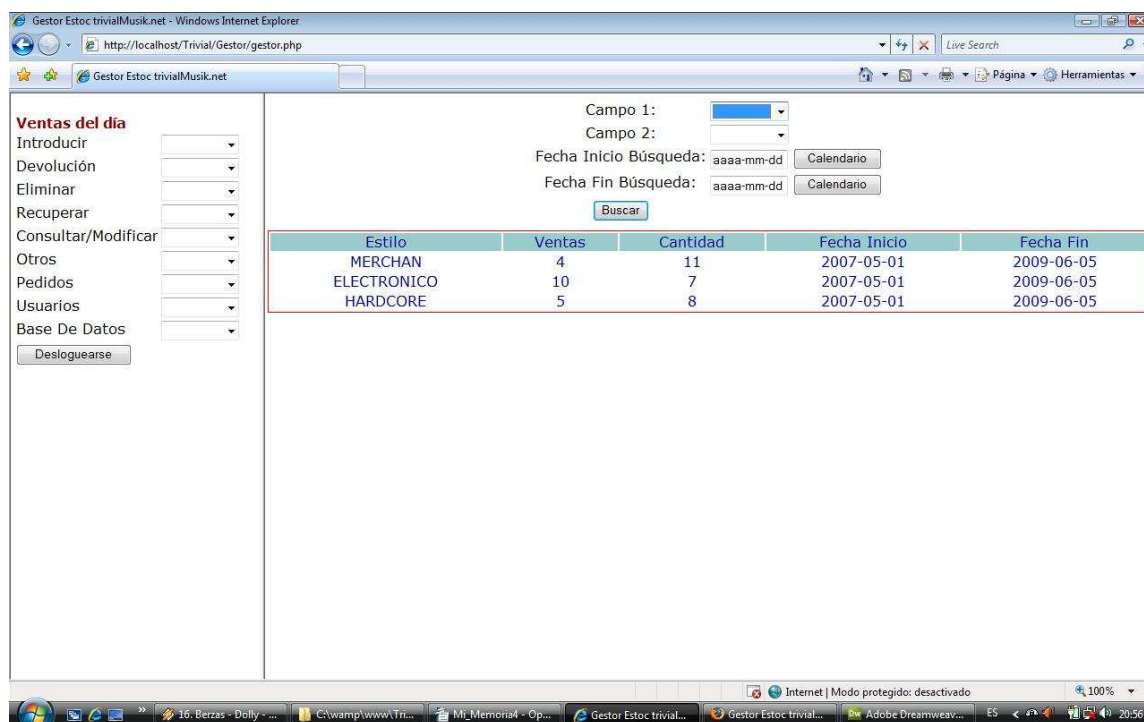
Propósito: Obtener listado de ventas según estilo en un periodo determinado.

Resumen: Manera de obtener listado de ventas, compras, pedidos y logs.

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Otros" > "Listados".	2- El sistema redirecciona al cliente a una página Web mostrando un formulario.
3- El usuario rellena el formulario y pulsa "Buscar".	4-El sistema carga una página dinámica con el resultado de la búsqueda.

Captura de pantalla de listado de ventas según estilo



Curso alternativo de los acontecimientos

Actores	Sistema
1- El usuario deja de rellenar alguna casilla del formulario.	2- El sistema muestra pop-up mostrando el error.

Caso de uso: Listar pedidos pendientes.

Actores: Administrador.

Tipo: Primario.

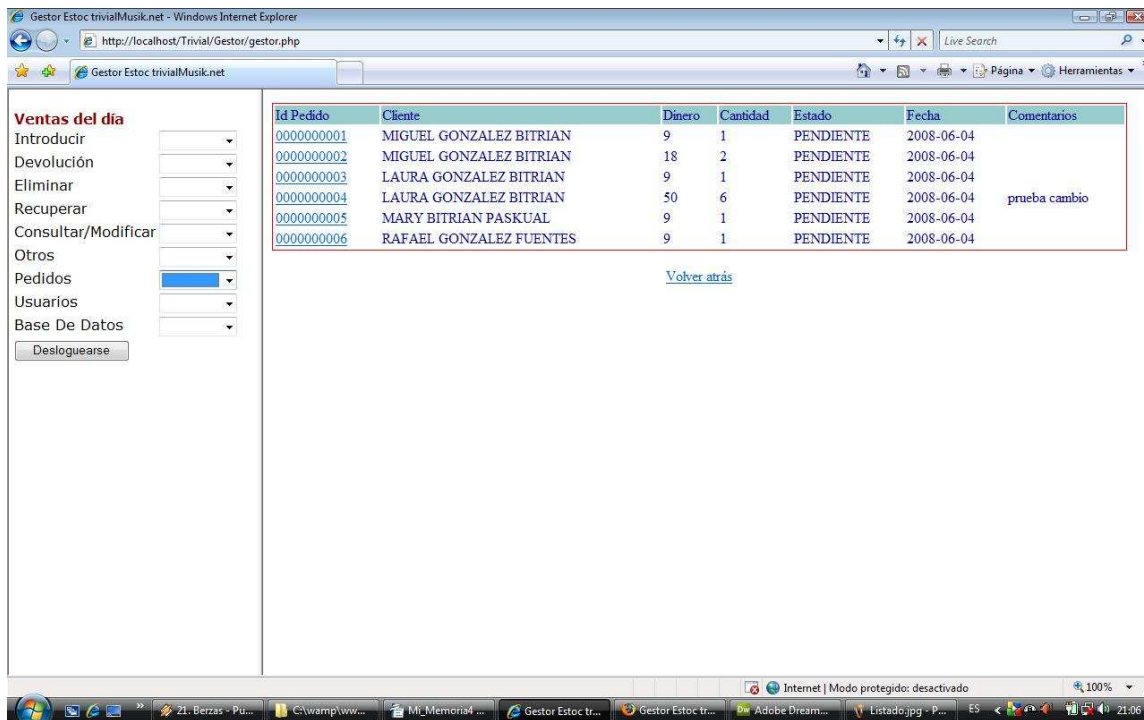
Propósito: Obtener listado de los pedidos on-line que están en estado pendiente.

Resumen: Manera listado de los pedidos según su estado.

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario pulsa en el menú "Pedidos" > "Pendientes".	2- El sistema redirecciona al cliente a una página Web mostrando un formulario.
3- El usuario puede ver el pedido clicando en el "Id Pedido"	

Captura de pantalla de los pedidos pendientes



2.5.2 Página Web

Caso de uso: Registro como nuevo usuario.

Actores: Administrador y cliente.

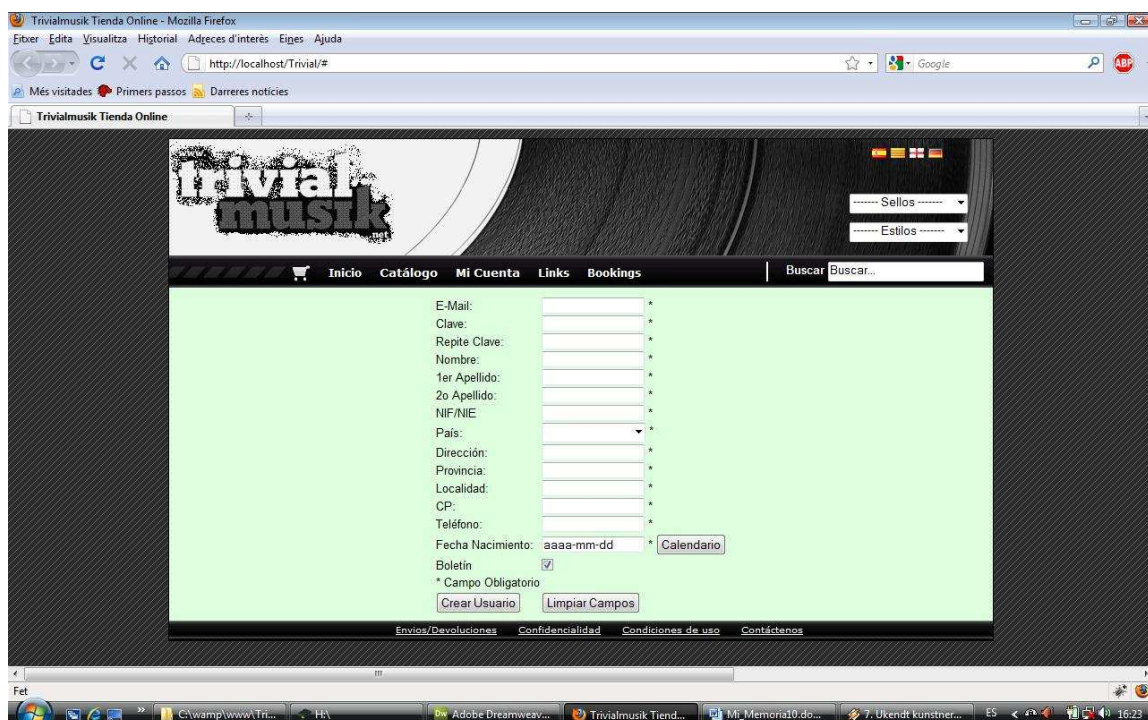
Tipo: Primario.

Propósito: Un cliente quiere registrarse para poder escuchar los vinilos, ver los precios y posteriormente poder hacer compras.

Resumen: Un cliente llega a la Web (<http://www.trivialmusik.net>), aún no está registrado en la Web. Clica en el menú en "Mi Cuenta" > "Crear Cuenta", rellena el formulario con sus datos personales (serán diferentes dependiendo de si se trata de un cliente minorista o mayorista), acepta las condiciones y salva sus datos.

Curso normal de los acontecimientos

Actores	Sistema
1- El usuario llega a la Web i clica en "Mi Cuenta" > "Crear Cuenta".	2- El sistema responde mostrando un formulario. Podrá elegir entre cliente minorista o cliente distribuidor, dependiendo de la opción escogida variará el formulario.
3- El usuario rellena el formulario con sus datos personales y acepta las condiciones.	4- El usuario guarda todos los datos en la base de datos y muestra una Web dinámica informativa donde se comunica al usuario que recibirá un e-mail para poder activar su cuenta.
5- El usuario accede a su e-mail personal y clica en el link de activación de cuenta.	6- El usuario puede acceder a la aplicación completa.



Caso de uso: Identificación de los usuarios registrados.

Actores: Administrador y cliente.

Tipo: Primario.

Propósito: Un cliente introduce su e-mail y su password para poder acceder a su cuenta, poder escuchar vinilos, consultar los precios y poder hacer compras.

Resumen: El cliente accede en el menú a "Mi Cuenta" > "Login/Logout", introduce los datos y accede al Web.

Curso normal de los acontecimientos

Actores	Sistema
1- El cliente accede en el menú a "Mi Cuenta" > "Login/Logout"	2- El sistema redirecciona al cliente al formulario.
3- El cliente introduce su e-mail y su clave	4- El sistema recoge los valores de los dos parámetros y verifica que el e-mail exista y que la clave sea correcta. Esta comprobación la realiza haciendo una consulta a la base datos.
	5- Si los datos son correctos se redirecciona al cliente a la página de inicio.



Curso alternativo de los acontecimientos

Actores	Sistema
1- El cliente accede en el menú a "Mi Cuenta" > "Login/Logout"	2- El sistema redirecciona al cliente al formulario.
3- El cliente introduce su e-mail y su clave	4- El sistema recoge los valores de los dos parámetros y comprueba que el e-mail no existe o la clave es incorrecta.
	5- Redirecciona al cliente a la página de login informando que ha introducido mal algún dato.

Caso de uso: Desloguearse de la Web un usuario registrados.

Actores: Administrador y cliente.

Tipo: Primario.

Propósito: El cliente no quiere seguir visitando o haciendo compras en la Web y no quiere que alguien pueda acceder a su cuenta.

Resumen: El cliente accede en el menú a "Mi Cuenta" > "Login/Logout".

Curso normal de los acontecimientos

Actores	Sistema
1- El cliente accede en el menú a "Mi Cuenta" > "Login/Logout"	2- El sistema redirecciona al cliente a la página Web de inicio.

Casos de uso: Modificación de los datos de los clientes.

Actores: Administrador y cliente.

Tipo: Primario.

Propósito: Sección para que los clientes puedan modificar sus datos personales.

Resumen: El cliente entra en la Web y quiere modificar sus datos personales.

Captura de pantalla inicio / deslogueo



Curso normal de los acontecimientos

Actores	Sistema
1- Este caso de uso comienza cuando un usuario ya está registrado y ha accedido a la página Web. Entonces selecciona la opción "Modificar" en el menú "Mi Cuenta".	2- El Sistema responde generando una página Web dinámica que contendrá el mismo formulario que cuando el cliente se registró pero relleno con sus datos personales.
3- El cliente modifica sus datos y hace click en la opción guardar.	4- El sistema responde modificando internamente todos los datos del cliente. Cierra la página dinámica con sus datos y abre la página principal.

Casos de uso: Búsqueda de artículos por sellos o estilos.

Actores: Administrador y cliente.

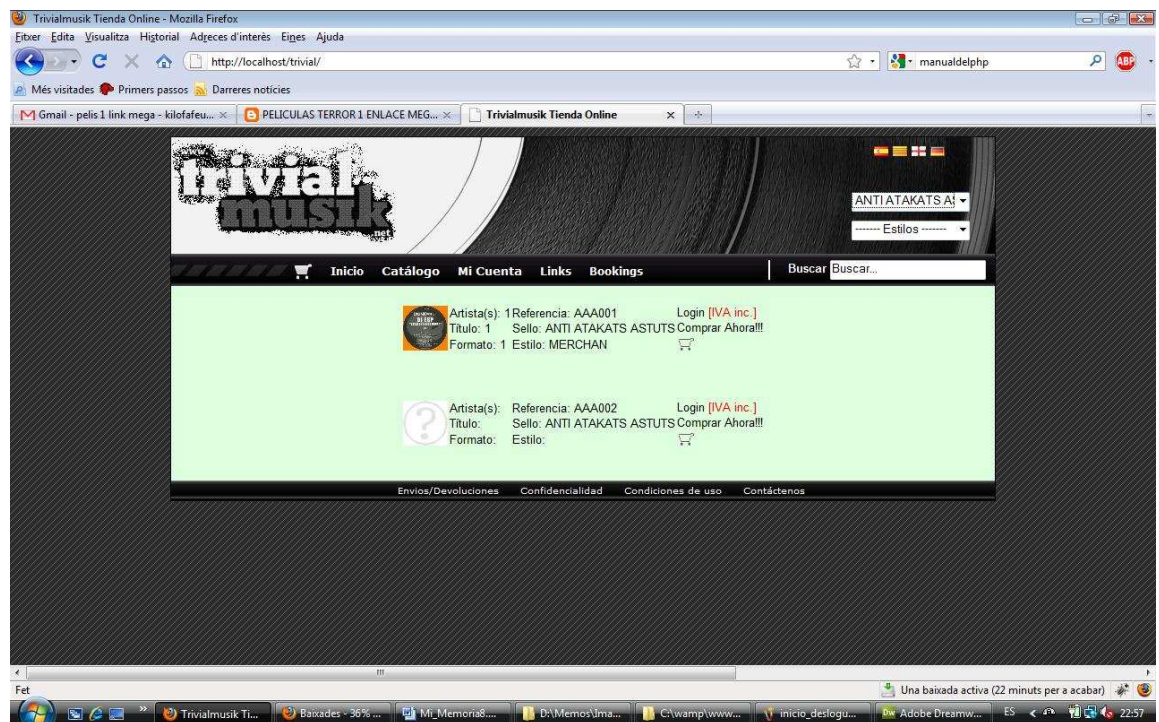
Tipo: Primario.

Propósito: Que el cliente pueda encontrar cualquier artículo que se disponga en la tienda.

Resumen: El cliente entra en la Web y hace una búsqueda por sellos o estilos.

Curso normal de los acontecimientos

Actores	Sistema
1- El cliente entra en la página Web y en la página principal clicla en el desplegable estilos o sellos y selecciona una de las opciones mostradas.	2- El Sistema responde generando una página Web dinámica que contiene los artículos que coincidan con la búsqueda hecha por el cliente.
3- El cliente puede acceder a la información de los artículos listados e introducirlos en el carrito de la compra, para poder consultar el precio y poder escuchar los vinilos debe estar logado. Si no lo está aparece un link en vez del precio del producto.	

Captura de pantalla de búsqueda de artículos

Casos de uso: Búsqueda de artículos según sean novedad, reposición u oferta.

Actores: Administrador y cliente.

Tipo: Primario.

Propósito: Que el cliente pueda encontrar cualquier artículo que se disponga en la tienda.

Resumen: El cliente entra en la Web y hace una búsqueda por novedad, reposición u oferta.

Curso normal de los acontecimientos

Actores	Sistema
1- El cliente entra en la página Web y en la página principal clic en el menú "Catálogo" y la opción que desee.	2- El Sistema responde generando una página Web dinámica que contiene los artículos que coincidan con la búsqueda hecha por el cliente.
3- El cliente puede acceder a la información de los artículos listados e introducirlos en el carrito de la compra, para poder consultar el precio y poder escuchar los vinilos debe estar logado. Si no lo está aparece un link en vez del precio del producto.	

Casos de uso: Búsqueda de artículos según coincidencia.

Actores: Administrador y cliente.

Tipo: Primario.

Propósito: Que el cliente pueda encontrar cualquier artículo que se disponga en la tienda.

Resumen: El cliente entra en la Web y hace una búsqueda por coincidencia, introduciendo la cadena de caracteres que desee encontrar en los artículos.

Curso normal de los acontecimientos

Actores	Sistema
1- El cliente entra en la página Web y en la página principal introduce en la casilla "Búsqueda" lo que quiere encontrar.	2- El Sistema responde generando una página Web dinámica que contiene los artículos que coincidan con la búsqueda hecha por el cliente.
3- El cliente puede acceder a la información de los artículos listados e introducirlos en el carrito de la compra, para poder consultar el precio y poder escuchar los vinilos debe estar logado. Si no lo está aparece un link en vez del precio del producto.	

Casos de uso: Visualización de un artículo.

Actores: Administrador y cliente.

Tipo: Primario.

Propósito: Poder visualizar las características del artículo así como poder escuchar el sample en caso de existir.

Resumen: El cliente entra en la Web y hace una búsqueda, encuentra un artículo que le interesa y desea ver sus características.

Curso normal de los acontecimientos

Actores	Sistema
1- Una vez realizada una búsqueda el cliente clicla sobre la imagen del artículo.	2- El Sistema responde generando una página Web dinámica muestra la imagen más grande, un reproductor de audio y las características del artículo.

Captura de pantalla visualización de un artículo



Casos de uso: Compra de artículos.

Actores: Administrador y cliente.

Tipo: Primario.

Propósito: Que el cliente pueda comprar los productos que tenga en su carrito de la compra.

Resumen: El cliente una vez logueado y con artículos en su carrito de la compra quiere comprar estos artículos

Curso normal de los acontecimientos

Actores	Sistema
1- El cliente (una vez logueado y con artículos en carrito de la compra) clic en el icono de carrito en el menú principal.	2- El Sistema responde generando una página Web dinámica que muestra los artículos de que se dispone y el total de la venta.
3- El cliente continúa con la compra pulsando "Comprar". El cliente aquí también podría modificar sus preferencias.	3- El sistema responde generando una factura e informando que en brevedad se pondrán en contacto con el cliente.

Casos de uso: Cambio de idioma de la página Web.

Actores: Administrador y cliente.

Tipo: Secundario.

Propósito: Que el cliente pueda visitar la página en el idioma en el que le sea más cómodo.

Resumen: El cliente quiere ver en un idioma más usable para él.

Curso normal de los acontecimientos

Actores	Sistema
1- El cliente en la página Web clic en una de las banderas que corresponde a un idioma (situadas arriba a la derecha)	2- El Sistema responde generando la página Web inicial en el idioma seleccionado. Todas las visitas a partir de este momento se verán en el idioma seleccionado.

2.6. Funcionamiento dinámico de la aplicación

Una aplicación Web es un conjunto de páginas Web estáticas y dinámicas. Una página Web estática es aquella que no cambia cuando un usuario la solicita: el servidor Web envía la página al navegador Web solicitante sin modificarla. Por el contrario, el servidor modifica las páginas Web dinámicas antes de enviarlas al navegador solicitante. El hecho cambiante de este tipo de página es la que le da el nombre de dinámica.

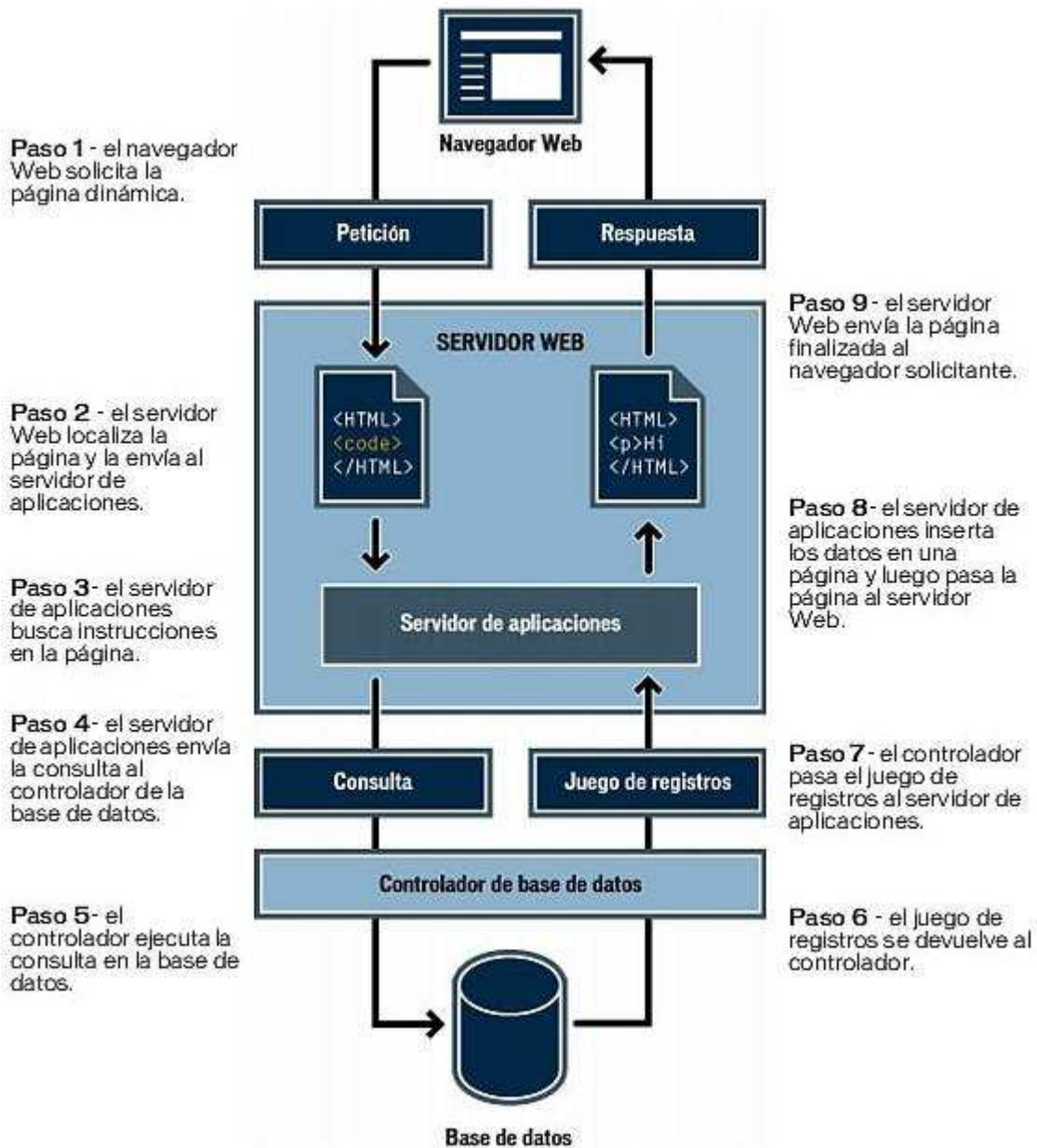
Cuando un servidor Web recibe una petición para mostrar una página Web estática, el servidor la envía directamente al navegador que la solicita. Cuando el servidor Web recibe una petición para mostrar una página dinámica, sin embargo, reacciona de forma distinta: transfiere la página a un software especial encargado de finalizar la página. Este software especial se denomina servidor de aplicaciones.

El servidor de aplicaciones lee el código de la página, finaliza la página en función de las instrucciones del código y elimina el código de la página. El resultado es una página estática que el servidor de aplicaciones devuelve al servidor Web, que a su vez la envía al navegador solicitante. Lo único que el navegador recibe cuando llega la página es código HTML puro.

Además de esto hay que tener en cuenta que un servidor de aplicaciones permite trabajar con recursos del lado del servidor, como las bases de datos. Por ejemplo, una página dinámica puede indicar al servidor de aplicaciones que extraiga datos de una base de datos y los inserte en el código HTML de la página. La instrucción para extraer datos de una base de datos recibe el nombre de. Una consulta consta de criterios de búsqueda expresados en un lenguaje de base de datos denominado SQL (Structured Query Language, lenguaje de consulta estructurado). La consulta SQL se escribe en los scripts o etiquetas del lado del servidor de la página.

Un servidor de aplicaciones no se puede comunicar directamente con una base de datos porque el formato de esta última impide que se descifren los datos. El servidor de aplicaciones sólo se puede comunicar con la base de datos a través de un controlador que actúe de intermediario con la base de datos: el software actúa entonces como un intérprete entre el servidor de aplicaciones y la base de datos.

Una vez que el controlador establece la comunicación, la consulta se ejecuta en la base de datos y se crea un juego de registros. Un juego de registros es un conjunto de datos extraídos de una o varias tablas de una base de datos. El juego de registros se devuelve al servidor de aplicaciones, que emplea los datos para completar la página.



3. Análisis y Diseño de la Base de Datos

El diseño de una base de datos es un proceso complejo que abarca decisiones a muy distintos niveles. La complejidad se controla mejor si se descompone el problema en subproblemas y se resuelve cada uno de estos subproblemas independientemente, utilizando técnicas específicas. Así, el diseño de una base de datos se descompone en diseño conceptual, diseño lógico y diseño físico.

3.1. Diseño Conceptual

El diseño conceptual parte de las especificaciones de requisitos de usuario y su resultado es el esquema conceptual de la base de datos. Un esquema conceptual es una descripción de alto nivel de la estructura de la base de datos, independientemente del SGBD que se vaya a utilizar para manipularla. Un modelo conceptual es un lenguaje que se utiliza para describir esquemas conceptuales. El objetivo del diseño conceptual es describir el contenido de información de la base de datos y no las estructuras de almacenamiento que se necesitarán para manejar esta información.

3.1.1. Diagrama Chen

Propuesto por Chen a mediados de los años setenta como medio de representación conceptual de los problemas y para representar la visión de un sistema de forma global. Físicamente adopta la forma de un grafo escrito en papel al que se denomina diagrama Entidad-Relación. Sus elementos fundamentales son las entidades y las relaciones.

Una entidad caracteriza a un tipo de objeto, real o abstracto, del problema a modelar. Toda entidad tiene existencia propia, es distinguible del resto de las entidades, tiene nombre y posee atributos definidos en un dominio determinado. Una entidad es todo aquello de lo que se desea almacenar información. En el diagrama E-R las entidades se representan mediante rectángulos.

Una relación es una asociación o relación matemática entre varias entidades. Las relaciones también se nombran. Se representan en el diagrama E-R mediante flechas y rombos. Cada entidad interviene en una relación con una determinada cardinalidad. La cardinalidad (número de instancias o elementos de una entidad que pueden asociarse a un elemento de la otra entidad relacionada) se representa mediante una pareja de datos, en minúsculas, de la forma (*cardinalidad mínima*, *cardinalidad máxima*), asociada a cada uno de las entidades que intervienen en la relación. Son posibles las siguientes cardinalidades: $(0,1)$, $(1,1)$, $(0,n)$, $(1,n)$, (m,n) . También se informa de las cardinalidades máximas con las que intervienen las entidades en la relación.

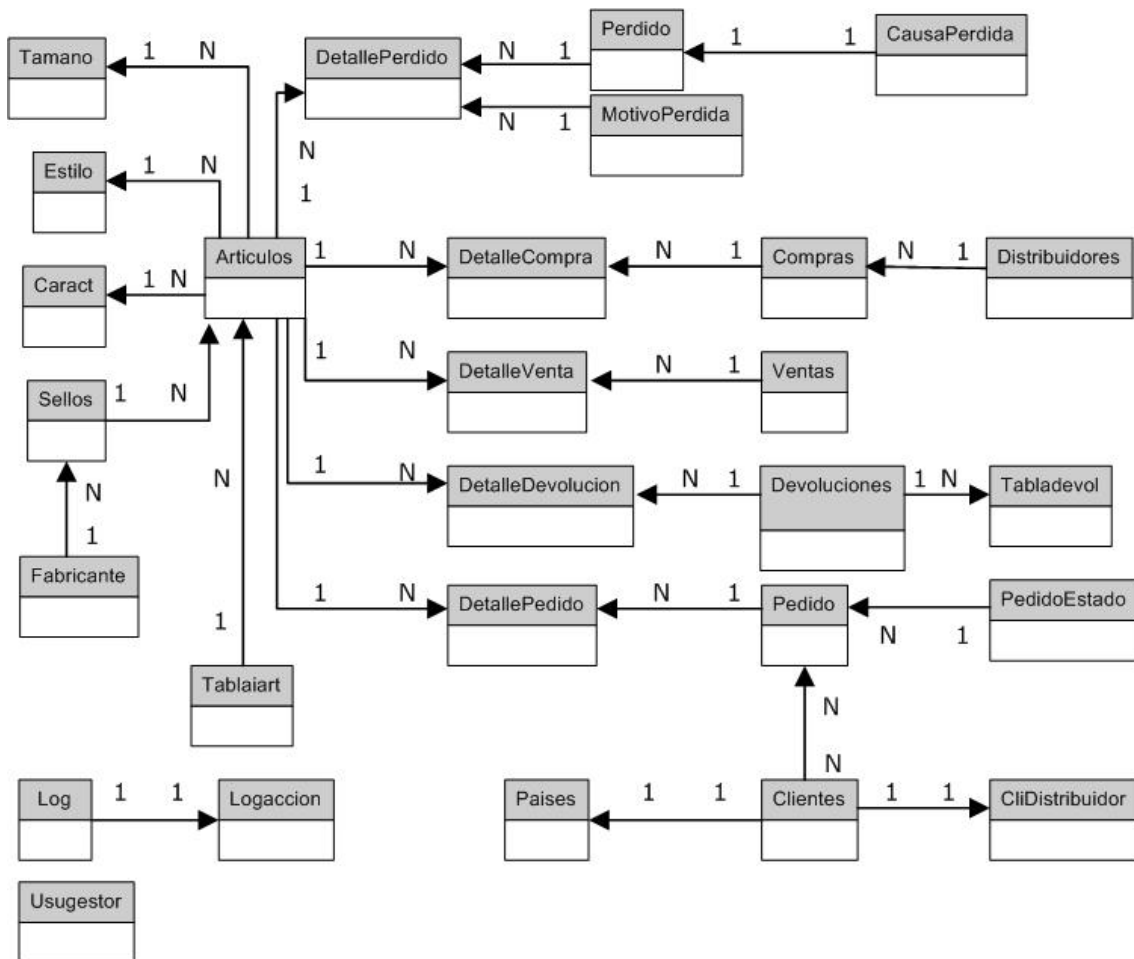
El tipo de relación se define tomando los máximos de las cardinalidades que intervienen en la relación. Hay cuatro tipos posibles:

- Una a una (1:1). En este tipo de relación, una vez fijado un elemento de una entidad se conoce la otra. Ejemplo: nación y capital.
- Una a muchas (1:N). Ejemplo: cliente y pedidos.

- Muchas a una (N:1). Simetría respecto al tipo anterior según el punto de vista de una u otra entidad.
- Muchas a muchas (N:N). Ejemplo: personas y viviendas.

Toda entidad debe ser unívocamente identificada y distinguible mediante un conjunto de atributos (quizás un solo atributo) denominado identificador o clave principal o primaria. Puede haber varios posibles identificadores para una misma entidad, en cuyo caso se ha de escoger uno de ellos como identificador principal siendo el resto identificadores alternativos. Ejemplo: DNI y número de seguridad social de una persona.

Hay unas normas de sentido común a seguir cuando se dibuja un diagrama E-R. La primera es emplear preferentemente líneas rectas en las relaciones y evitar en lo posible que estas líneas se crucen. Se suele usar nombres para describir las entidades y verbos para las relaciones. Esto es lógico ya que las entidades se ponen en común cuando se realiza alguna acción. Los verbos empleados no necesariamente tienen que ser siempre infinitivos.



3.2. Diseño Lógico y Diseño Físico de la Base de Datos

El diseño lógico parte del esquema conceptual y da como resultado un esquema lógico. Un esquema lógico es una descripción de la estructura de la base de datos en términos de las estructuras de datos que puede procesar un tipo de SGBD. Un modelo lógico es un lenguaje usado para especificar esquemas lógicos (modelo relacional, modelo de red, etc.). El diseño lógico depende del tipo de SGBD que se vaya a utilizar, no depende del producto concreto.

El diseño físico parte del esquema lógico y da como resultado un esquema físico. Un esquema físico es una descripción de la implementación de una base de datos en memoria secundaria: las estructuras de almacenamiento y los métodos utilizados para tener un acceso eficiente a los datos. Por ello, el diseño físico depende del SGBD concreto y el esquema físico se expresa mediante su lenguaje de definición de datos.

A partir de la tabla de entidades y atributos y la del diagrama Chen del análisis podemos crear las tablas de nuestro sistema.

En las siguientes tablas los campos subrayados indican que son claves primarias ("primary keys"), y los campos en cursiva indican que son claves foráneas ("foreign keys")

Estructura de la tabla artículos:

En esta tabla se guardan todos los artículos de la tienda, sea cual sea su tipo y todas sus características.

Cuando el artículo sea de merchandising o material electrónico los campos Disco, Id_Tamaño estarán vacíos, a null.

La tabla artículos se escribe sin tilde para evitar problemas a la hora de hacer búsquedas, ya que a veces se pueden guardar con caracteres extraños.

Campo	Tipo	Nulo	Defecto
<u>Id_Articulo</u>	int(10)	Si	NULL
RefNom	varchar(10)	Si	
RefNum	int(3)	Si	000
Artista	varchar(30)	Si	
Disco	varchar(30)	Si	
PVPMenor	float(4,2)	Si	0.00
PVPDistr	float(4,2)	Si	0.00
Cantidad	int(4)	Si	0
<u>Id_Caract</u>	tinyint(1)	Si	NULL
<u>Id_Tamano</u>	tinyint(2)	Si	NULL
<u>Id_Estilo</u>	tinyint(2)	Si	NULL
Fecha1	date	Si	0000-00-00
FechaUlt	date	Si	0000-00-00
Imagen	varchar(200)	Si	NULL
Descripcion	varchar(50)	Si	NULL
Tracklist	varchar(100)	Si	NULL
MP3	varchar(200)	Si	NULL
Borrado	tinyint(1)	Si	0

Estructura de la tabla caract:

Esta tabla nos marca si el artículo es novedad, está en oferta, está descatalogado o ninguna de las anteriores. Su codificación es:

- 1 Normal
- 2 Novedad
- 3 Oferta
- 4 Descatalogado

Campo	Tipo	Nulo	Defecto
Id_Caract	tinyint(1)	Si	NULL
TipoCaract	varchar(13)	Si	
Imagen	blob	Si	NULL

Estructura de la tabla causaperdida:

En esta tabla se proporciona el número de identificador de artículo perdido y se guardan las referencias para tal identificador.

Campo	Tipo	Nulo	Defecto
Id_Perdida	int(8)	Si	NULL
RefNom	varchar(10)	Si	
RefNum	int(3)	Si	000

Estructura de la tabla clidistribuidor:

En esta tabla se guardan los datos de todos los clientes mayoristas, es decir, aquí se guardan los datos de la tienda a la que se le venden los vinilos

Campo	Tipo	Nulo	Defecto
Id_Cliente	int(10)	Si	NULL
CIF	blob	Si	
NombreTi	blob	Si	
DireccTi	blob	Si	
TelfTi	blob	Si	
FAXTi	blob	Si	
MailTi	blob	Si	
WebTi	blob	Si	

Estructura de la tabla clientes:

En esta tabla se guardan los datos generales de cada cliente.

Campo	Tipo	Nulo	Defecto
Id_Cliente	int(10)	Si	NULL
Mail	blob	Si	
Nombre	varchar(15)	Si	
Apellido1	varchar(15)	Si	
Apellido2	varchar(15)	Si	
DNI	blob	Si	
Pais	blob	Si	
Direccion	blob	Si	
Provincia	blob	Si	
Localidad	blob	Si	
CP	blob	Si	
Telf	blob	Si	
FechaNac	blob	Si	
Boletin	tinyint(4)	Si	0
Clave	blob	Si	NULL
TipoCliente	tinyint(1)	Si	0

Estructura de la tabla compras:

Esta tabla proporciona un identificador de compra, se guarda la fecha y la distribuidora a la que se le han comprado el/los artículo/s.

Campo	Tipo	Nulo	Defecto
Id_Compra	int(10)	Si	NULL
Fecha	date	Si	0000-00-00
Id_Distribuidor	tinyint(3)	Si	

Estructura de la tabla detallecompra:

Esta tabla relaciona el identificador de compra a las referencias que se hayan adquirido así como la cantidad y el precio.

Campo	Tipo	Nulo	Defecto
Id_detallecompra	bigint(20)	Si	NULL
Id_Compra	int(10)	Si	
RefNom	varchar(10)	Si	
RefNum	int(3)	Si	000
Cantidad	int(4)	Si	0
PrecioCompra	float(4,2)	Si	0.00

Estructura de la tabla detalledevolucion:

Esta tabla proporciona un detalle de los artículos devueltos relacionándolos mediante el identificador de devolución.

Campo	Tipo	Nulo	Defecto
Id_detalledevolucion	bigint(20)	Si	NULL
Id_Devolucion	int(10)	Si	
RefNom	varchar(10)	Si	
RefNum	int(3)	Si	000
Cantidad	int(4)	Si	0

Estructura de la tabla detallepedido:

Esta tabla proporciona un detalle de los artículos comprados relacionándolos mediante el identificador de pedido.

Campo	Tipo	Nulo	Defecto
Id_DetallePedido	int(10)	Si	NULL
Id_Pedido	int(10)	Si	
Id_Articulo	int(10)	Si	
Cantidad	int(4)	Si	
Distribuidor	tinyint(1)	Si	0

Estructura de la tabla detalleperdido:

Esta tabla proporciona un detalle de los artículos devueltos relacionándolos mediante el identificador de devolución.

Campo	Tipo	Nulo	Defecto
Id_detalleperdido	bigint(20)	Si	NULL
Id_Perdida	int(10)	Si	
RefNom	varchar(10)	Si	
RefNum	int(3)	Si	000
Cantidad	int(4)	Si	0
Id_Motivo	tinyint(2)	Si	

Estructura de la tabla detalleventa:

Esta tabla relaciona el identificador de venta a las referencias adquiridas por el cliente. También guarda la cantidad adquirida y el precio al que se le vendió.

Campo	Tipo	Nulo	Defecto
Id_detalleventa	bigint(20)	Si	NULL
Id_Venta	int(10)	Si	
Id_Cliente	int(10)	Si	0000000000
RefNom	varchar(10)	Si	
RefNum	int(3)	Si	000
Cantidad	int(4)	Si	0
PrecioVenta	float(4,2)	Si	0.00
Descuento	float(4,2)	Si	0.00

Estructura de la tabla devoluciones:

Esta tabla proporciona un identificador de devolución y guarda la fecha así como el distribuidor.

Campo	Tipo	Nulo	Defecto
Id_Devolucion	int(10)	Si	NULL
Fecha	date	Si	0000-00-00
Id_Distribuidor	tinyint(3)	Si	

Estructura de la tabla distribuidores:

En esta tabla se guardan los identificadores de los distribuidores de la tienda con sus datos.

Campo	Tipo	Nulo	Defecto
Id_Distribuidor	tinyint(3)	Si	NULL
NombreDistr	varchar(25)	Si	
Direc	varchar(50)	Si	
CP	varchar(10)	Si	
Ciudad	varchar(20)	Si	
Pais	varchar(15)	Si	
Web	varchar(50)	Si	
NomCont1	varchar(20)	Si	NULL
Telf1	bigint(14)	Si	NULL
Email1	varchar(50)	Si	NULL
FAX1	bigint(14)	Si	NULL
NomCont2	varchar(20)	Si	NULL
Telf2	bigint(14)	Si	NULL
Email2	varchar(50)	Si	NULL
FAX2	bigint(14)	Si	NULL
Borrado	tinyint(1)	Si	0

Estructura de la tabla estilo:

Esta tabla describe de que estilo es el artículo almacenado, dado que se pueden ir introduciendo nuevos estilos en la tienda se ha optado por hacerlo de esta manera. La codificación inicial es:

- 01 Merchan
- 02 Electrónico
- 03 Schranz
- 04 HardTechno
- 05 HardTek
- 06 Hardcore
- 07 D&B
- 08 RaggaJungle
- 09 Techno
- 10 Acid

Campo	Tipo	Nulo	Defecto
Id_Estilo	tinyint(2)	Si	NULL
TipoEstilo	varchar(11)	Si	

Estructura de la tabla fabricante:

Esta tabla almacena los fabricantes, el campo Borrado sirve para indicar si el fabricante sigue activo o no.

Campo	Tipo	Nulo	Defecto
Id_fabricante	tinyint(4)	Si	NULL
Fabricante	varchar(50)	Si	
Borrado	tinyint(1)	Si	0

Estructura de la tabla log:

Esta tabla guarda las acciones realizadas por los usuarios en el gestor, de manera que siempre queden registradas las acciones realizadas.

Campo	Tipo	Nulo	Defecto
id_log	bigint(20)	Si	NULL
id_accion	tinyint(3)	Si	
usuario	varchar(50)	Si	
fecha	date	Si	
hora	time	Si	

Estructura de la tabla logaccion:

Esta tabla almacena la descripción de las acciones que pueden realizar los usuarios del gestor.

Campo	Tipo	Nulo	Defecto
id_accion	tinyint(3)	Si	NULL
accion	varchar(50)	Si	

Estructura de la tabla motivoperdida:

En esta tabla se codifican los posibles motivos de la pérdida, se hace así para poder ir añadiendo nuevos motivos en caso de que aparezcan.

- 01 Rallado
- 02 Doblado
- 03 Robado
- 04 Agujereado
- 05 Manchado
- 06 Golpeado

Campo	Tipo	Nulo	Defecto
Id_Motivo	tinyint(2)	Si	NULL
Causa	varchar(15)	Si	

Estructura de la tabla países:

Esta tabla relaciona los países del mundo con un identificador.

Campo	Tipo	Nulo	Defecto
id	int(3)	Si	0
pais	varchar(30)	Si	

Estructura de la tabla pedido:

Almacena los pedidos realizados por Internet.

Campo	Tipo	Nulo	Defecto
Id_Pedido	int(10)	Si	NULL
Id_Estado	tinyint(1)	Si	1
Id_Cliente	int(10)	Si	
DineroPedido	int(5)	Si	0
Cantidad	tinyint(4)	Si	
FechaR	date	Si	0000-00-00
Comentario	varchar(100)	Si	

Estructura de la tabla pedidoestado:

Esta tabla proporciona una descripción del estado del pedido:

Campo	Tipo	Nulo	Defecto
Id_Estado	tinyint(1)	Si	NULL
Estado	varchar(9)	Si	

Estructura de la tabla perdido:

En esta tabla se guarda la fecha y la cantidad de artículos perdidos.

Campo	Tipo	Nulo	Defecto
Id_Perdida	int(10)	Si	NULL
Fecha	date	Si	0000-00-00

Estructura de la tabla sellos:

En esta tabla se guarda un listado de todos los sellos discográficos de los que se ha tenido algún artículo. Se guardará el nombre completo del sello, su referencia, mail del sello y Web si los poseen.

Campo	Tipo	Nulo	Defecto
Id_Sello	int(10)	Si	NULL
RefNom	varchar(10)	Si	
NombreSello	varchar(20)	Si	
Id_fabricante	tinyint(4)	Si	NULL
Dueno	varchar(50)	Si	
Mail1	varchar(50)	Si	
Mail2	varchar(50)	Si	
Web1	varchar(50)	Si	
Web2	varchar(50)	Si	
Telf1	bigint(14)	Si	0
Telf2	bigint(14)	Si	0
Borrado	tinyint(1)	Si	0

Estructura de la tabla tabladefectuoso:

Esta tabla almacena los artículos defectuosos relacionando el artículo con la causa y el motivo.

Campo	Tipo	Nulo	Defecto
Id_Perdido	int(3)	Si	NULL
Causa	varchar(15)	Si	
Id_Motivo	tinyint(2)	Si	
Referencia	varchar(15)	Si	
Cantidad	int(4)	Si	0

Estructura de la tabla tabladevol:

Esta tabla guarda las devoluciones que realizamos a nuestros distribuidores, guarda la referencia del artículo la cantidad y el distribuidor

Campo	Tipo	Nulo	Defecto
Id_Devol	int(10)	Si	NULL
Referencia	varchar(13)	Si	
Cantidad	int(4)	Si	
Distribuidor	tinyint(3)	Si	

Estructura de la tabla tablaiart:

Esta tabla se utiliza a la hora de introducir un artículo en la base de datos, esta tabla es necesaria ya que para introducir artículos se utiliza AJAX, por lo que es necesario gestionarlo para poder hacer una subida masiva de artículos.

Campo	Tipo	Nulo	Defecto
Id_Articulo	bigint(10)	Si	NULL
Referencia	varchar(13)	Si	
RefNom	varchar(10)	Si	
RefNum	int(3)	Si	000
Artista	varchar(30)	Si	
Disco	varchar(30)	Si	
PVPMenor	float(4,2)	Si	0.00
PVPDistr	float(4,2)	Si	0.00
Cantidad	int(4)	Si	0
Id_Caract	tinyint(1)	Si	NULL
Id_Tamano	tinyint(2)	Si	NULL
Id_Estilo	tinyint(2)	Si	NULL
Imagen	blob	Si	NULL
Descripcion	varchar(50)	Si	NULL
Tracklist	varchar(100)	Si	NULL
MP3	mediumblob	Si	NULL

Estructura de la tabla tablaicomp:

Esta tabla se utiliza para introducir las compras realizadas a la base datos. Es necesaria para poder utilizar AJAX a la hora de archivar las compras.

Campo	Tipo	Nulo	Defecto
Referencia	varchar(13)	Si	
PrecioCompra	float(4,2)	Si	
Cantidad	int(4)	Si	
Distribuidor	tinyint(3)	Si	

Estructura de la tabla tablaisello:

Esta tabla se utiliza para introducir los sellos discográficos en la base de datos, es necesaria para poder introducir varios sellos de una sola vez.

Campo	Tipo	Nulo	Defecto
RefNom	varchar(10)	Si	
NombreSello	varchar(20)	Si	
Id_fabricante	tinyint(4)	Si	NULL
Dueno	varchar(50)	Si	
Mail1	varchar(20)	Si	
Mail2	varchar(20)	Si	
Web1	varchar(20)	Si	
Web2	varchar(20)	Si	
Telf1	bigint(14)	Si	0
Telf2	bigint(14)	Si	0

Estructura de la tabla tablaventa:

Esta tabla se utiliza a la hora de hacer las ventas, tanto desde la tienda como por Internet, en caso de ser necesaria hacer una factura esta tabla será base para ello.

Campo	Tipo	Nulo	Defecto
Referencia	varchar(13)	Si	
Precio	float(4,2)	Si	
Cantidad	int(4)	Si	
Total	float(7,2)	Si	
Descuento	float(4,2)	Si	0.00
TipoCliente	tinyint(1)	Si	0
Id_Cliente	int(10)	Si	0000000001

Estructura de la tabla tamaño:

Esta tabla describe de qué tamaño es el artículo guardado, su codificación inicial es:

01 12"
 02 Otro: Para indicar que no se trata de un vinilo o un CD
 03 CD
 04 12" x2
 05 12" x3
 06 12" x4
 07 12" x5
 08 10"
 09 10" x2
 10 10" x3
 11 7"
 12 7" x2

Campo	Tipo	Nulo	Defecto
Id_Tamano	int(1)	Si	NULL
TipoTamano	varchar(10)	Si	

Estructura de la tabla usugestor:

Esta tabla guarda el e-mail así como la clave para que los usuarios gestor puedan acceder al sistema.

Campo	Tipo	Nulo	Defecto
email	varchar(50)	Si	
clave	blob	Si	

Estructura de la tabla ventas:

Esta tabla proporciona un identificador de venta, es necesario proporcionarle un identificador de cliente, por lo que si la venta se realiza en la tienda el identificador serán todo ceros.

Campo	Tipo	Nulo	Defecto
Id_Venta	int(10)	Si	NULL
Id_Cliente	int(10)	Si	0000000000
Fecha	date	Si	0000-00-00
TipoCliente	tinyint(1)	Si	0

3.3. Tipos de datos utilizados

TINYINT: Es un entero muy pequeño. Su rango es de -128 a 127. Para enteros sin signo el rango es de 0 a 255.

BIT y BOOL: Es un sinónimo para TINYINT (1).

INT: Es un entero normal. Su rango es de -2147483648 a 2147483647. Para enteros sin signo el rango es de 0 a 4294967295.

SMALLINT: Un entero pequeño. Su rango es de -32768 a 32767. Para enteros sin signo el rango es de 0 a 65535.

FLOAT [(M, D)]: Un número pequeño de punto flotante (o bien, de precisión sencilla). El rango de valores permitidos es de -3.402823466E+38 a -1.175494351E-38, 0, y de 1.175494351E-38 a 3.402823466E+38. Si se usa el modificador UNSIGNED, los valores negativos son deshabilitados. M es el tamaño de despliegue y D el número de decimales. FLOAT sin argumentos o FLOAT(X) donde $X \leq 24$ se entiende como un número de punto flotante de precisión sencilla.

VARCHAR (M) [BINARY]: Una cadena de longitud variable. Los espacios sobrantes son removidos cuando el valor es almacenado. El rango de M es de 0 a 255 caracteres. Los valores VARCHAR son ordenados y comparados en modo no sensitivo a menos que se use el modificador BINARY.

BLOB ó TEXT: Una columna BLOB o TEXT con una longitud máxima de 65535 ($2^{16} - 1$) caracteres.

DATE: Valores de fecha. El rango válido es de '1000-01-01' a '9999-12-31'. MySQL muestra valores DATE en el formato 'YYYY-MM-DD', y permite asignar valores a columnas DATE usando cadenas o números.

Nota:

Hay que denotar que en MySQL si una columna en una tabla tiene una longitud variable, la fila completa se convierte en una de longitud variable. Del mismo modo, si una tabla contiene alguna columna de longitud variable (VARCHAR, TEXT o BLOB), todas las columnas CHAR de más de cuatro caracteres se cambian a columnas VARCHAR. Esto no afecta al modo en que se usen estas columnas; en MySQL, VARCHAR es sólo una forma diferente de almacenar caracteres. MySQL realiza esta conversión porque esto ahorra espacio y hace las operaciones en la tabla más rápidas.

A partir de MySQL 4.1.0, una columna CHAR o VARCHAR con una especificación de longitud mayor de 255 se convierte al tipo TEXT más pequeño que pueda contener valores de la longitud dada. Por ejemplo, VARCHAR (500) se convierte a TEXT, y VARCHAR (200000) se convierte a MEDIUMTEXT. Esto es un comportamiento para compatibilidad.

4. Diseño de la interficie gráfica

Los objetivos de diseño se definen como las restricciones que inciden en determinados aspectos de la calidad del software. En este caso, se han tenido en cuenta los siguientes:

1. Facilidad de uso: proporcionar una aplicación sencilla, intuitiva que no presente ninguna dificultad de uso para cualquier posible usuario de la misma, tanto para usuarios del gestor de stock como usuarios de la Web.
2. Facilidad de mantenimiento: ofrecer la posibilidad de mantener la aplicación de una manera sencilla y eficaz.
3. Futuras ampliaciones: con el fin de ampliar el sistema, el diseño debe ser estructurado y modular haciendo de esta tarea una mecanismo sencillo realizándolo en el menor tiempo posible.

4.1. Gestor de Stock

En un principio el gestor de stock se iba a realizar en VisualBasic.Net, iba a ser una aplicación de Windows, pero a la hora de intentar relacionar el gestor con las bases de datos surgieron algunos problemas. El problema más importante era que es muy difícil relacionar una base de datos en un PC con una base de datos en servidor, de manera que si algún campo se actualiza en alguna de las dos bases de datos automáticamente se modifica en la otra. Esto se quería hacer para poder controlar eficazmente y en todo momento la cantidad en stock de cada producto, tanto en la tienda como en la Web. Debido a la dificultad a realizar esta operación finalmente se optó por realizar el gestor de stock en Web y el generador de contratos, la opción de inventariar y la posibilidad de imprimir etiquetas como una aplicación Windows. De esta manera se ha ofrecido la posibilidad de poder actualizar todo el stock desde sitios remotos a la tienda, como por ejemplo desde casa de los trabajadores.

4.1.1. Inicio

Nos dirige a la pantalla de inicio del gestor, en esta pantalla aparece el logo de la empresa.

4.1.2. Ventas del día

Se puede acceder a esta función desde el menú principal ya que va a ser la función más utilizada.

Lo primero que hay que seleccionar a la hora de hacer una venta es si se trata de una venta al por menor o a un distribuidor, por defecto es una venta al por menor ya que la tienda está enfocada, en un principio, a particulares. Esto es importante ya que según la opción seleccionada se mostrará un precio de venta.

La siguiente opción a seleccionar es si se trata de una venta hecha vía Web o no, por defecto será que no. Cuando se hace una venta en Internet la tienda recibe un mail con el Id_Cliente y el Id_Venta, de esta manera se pueden buscar fácilmente en la base de datos artículos los artículos de dicha venta así como los datos del cliente para hacerla efectiva.

A la hora de introducir los artículos de una venta se procede de la siguiente manera. Para cada artículo introduciremos su referencia, formada por caracteres y un número, la cantidad de artículos (por defecto 1) y el descuento (por defecto será 0). Según vayamos pulsando caracteres en el campo referencia el programa debe mostrar posibles artículos a seleccionar, de forma que si se selecciona alguno de ellos el programa introducirá el nombre del artista y del disco. Para introducir el artículo en nuestra cuenta habrá que pulsar el botón "introducir"; al hacer esto se debe comprobar si el artículo existe en la base de datos, si no existe se debe mostrar un mensaje de error.

Si se introduce una cantidad negativa se estará marcando que es un artículo defectuoso, que no se puede vender. Cuando se hace esto aparece un campo (Motivo), este campo es un desplegable cargado de la base de datos ARTICULOS e indica que es lo que le ha sucedido al artículo. Todos los artículos defectuosos se guardan con su motivo en la base de datos. Si se introduce una cantidad negativa, hasta que se cierre esta "venta", todas las cantidades deberán ser negativas y tendrán el mismo motivo.

En este apartado, cuando se introduce la referencia, si el número no es de tres dígitos se rellenan los restantes (hasta tres) a cero. Además el texto se guarda en mayúsculas.

También existen dos botones de limpiar campos, uno que sólo limpia la búsqueda del artículo actual y otro que borra toda la venta actual, deja todos los campos en blanco. Debido a esto los artículos se quitan de la base de datos en el momento en que se cierre la venta pulsando el botón "Aceptar".

Otra opción existente es la de eliminar un artículo de la lista actual, es decir, si por equivocación se introduce un artículo que no se quiere vender se selecciona de la lista y se puede eliminar pulsando el botón "Eliminar Fila".

4.1.3. Gestor de stock

En esta parte del programa se controlan el resto de aspectos que corresponden al control de stocks:

4.1.4. Introducir Artículo

Esta opción permite introducir un nuevo artículo en la base datos. Hay que introducir el nombre y número de referencia, el precio de venta por menor y distribuidor, el nombre del artista, nombre del disco, tamaño, estilo, su característica, su audio (si lo tiene) y su imagen:

- Normal
- Novedad
- Oferta
- Descatalogado

El gestor comprueba que el sello (la referencia) existe en la base de datos. En caso de no existir debe mostrar un mensaje de error. También comprueba que el audio esté en formato "mp3", no supere los 2.048 Kb (2 Mb), esté grabado en mono y no supere los tres minutos y medio. La imagen por otro lado no debe ser más grande de 8 Kb.

4.1.5. Introducir Compra

En este apartado se introducen los artículos una vez comprados (no tienen por que estar aún en la tienda). Se debe introducir la referencia (nombre y número), el precio de compra, la cantidad adquirida y la distribuidora que lo envía.

A la hora de introducir una compra el gestor deberá comprobar que el sello (la referencia) existe en la tabla "Sellos", si no existe nos debe redirigir al formulario "Introducir/Modificar Sello".

Cuando esté instalado el escáner el programa debe preguntar si queremos imprimir etiquetas del artículo que estamos introduciendo y cuantas.

4.1.6. Introducir Sello

Esta opción permite un sello en nuestra base de datos.

Para introducir un sello el gestor debe solicitar su referencia, el nombre completo del sello, la dirección Web y un e-mail de contacto. Para verificar que el e-mail sea correcto se ha implementado una función en Javascript, en caso de no ser un e-mail válido aparece un mensaje de error.

4.1.7. Introducir Distribuidor

Esta opción permite introducir un distribuidor en nuestra base de datos.

Para introducir un distribuidor el gestor debe solicitar su nombre, dos teléfonos de contacto (uno es obligatorio), FAX de contacto (opcional), e-mail y Web.

4.1.8. Introducir Estilo

Esta opción permite introducir o modificar un estilo en nuestra base de datos.

Para introducir un estilo se debe introducir su nombre y pulsar el botón "Introducir". El gestor deberá comprobar que el estilo que se quiere introducir no exista.

4.1.9. Introducir Tamaño

Esta opción permite introducir un tamaño en nuestra base de datos.

Para introducir un tamaño se debe introducir su nombre y pulsar el botón "Introducir". El gestor deberá comprobar que el tamaño que se quiere introducir no exista.

4.1.10 Introducir Fabricante

Esta opción permite introducir un fabricante nuevo en nuestra base de datos.

Para introducir un fabricante se debe introducir su nombre y pulsar el botón "Introducir". El gestor deberá comprobar que el fabricante que se quiere introducir no exista.

4.1.11 Consultar/Modificar Artículo

Para poder consultar y/o modificar un artículo previamente introducido deberemos seleccionarlo de una lista desplegable o bien hacer una búsqueda por coincidencia, en este caso deberemos clicar en el artículo que deseamos ver. Entonces se nos mostrarán los datos del artículo de los cuales se podrán modificar todos menos su referencia. Aquí es donde podremos modificar los datos y guardar de nuevo el artículo después de hacer las comprobaciones necesarias.

4.1.12 Consultar/Modificar Sello

Para poder consultar y/o modificar un sello previamente introducido deberemos seleccionarlo de una lista desplegable o bien hacer una búsqueda por coincidencia, en este caso deberemos clicar en el sello que deseamos ver. Entonces se muestran los campos del sello. Si alguno es modificado guardan los cambios en la base de datos después de hacer las comprobaciones necesarias.

4.1.13 Consultar/Modificar Distribuidor

Para poder consultar y/o modificar un distribuidor previamente introducido deberemos seleccionarlo de una lista desplegable o bien hacer una búsqueda por coincidencia, en este caso deberemos clicar en el distribuidor que deseamos ver. Si se modifica, y después de hacer las comprobaciones necesarias, se guarda en la base de datos.

4.1.14 Consultar/Modificar Fabricante

Para poder consultar y/o modificar un fabricante previamente introducido deberemos seleccionarlo de una lista desplegable o bien hacer una búsqueda por coincidencia, en este caso deberemos clicar en el fabricante que deseamos ver. Si se modifica, y después de hacer las comprobaciones necesarias, se guarda en la base de datos.

4.1.15. Eliminar Referencia

Esta opción permite eliminar una referencia de la base de datos. Debe pedir la referencia y la confirmación. Esta opción sólo se puede realizar si no se han realizado ventas de la referencia, es decir, sirve para eliminar referencias mal introducidas.

4.1.16. Eliminar Sello

Esta opción permite eliminar un sello de la base de datos. Debe pedir la referencia y la confirmación. Esta opción sólo se puede realizar si no existen referencias en la base de datos "artículos".

4.1.17. Eliminar Distribuidor

Esta opción permite eliminar un distribuidor de la base de datos. Debe pedir el nombre del distribuidor y la confirmación. Esta opción sólo se puede realizar si no existen artículos comprados a este distribuidor.

4.1.18. Eliminar Estilo

Esta opción permite eliminar un estilo de la base de datos. Debe pedir el estilo y la confirmación. Esta opción sólo se puede realizar si no existen artículos con este estilo.

4.1.19. Listados

En este apartado se pueden hacer diferentes tipos de búsquedas en la base de datos. Se pueden hacer búsquedas por:

- Referencia
- Artista
- Distribuidora
- Tamaño
- Estilo
- Característica
- Cantidad en stock haciendo búsquedas en rangos de diez discos (0-9, 10-19, 20-29, 30-39, 40-50, > 50)
- Fecha de venta
- Fecha de compra
- Motivo pérdida

Se pueden hacer búsquedas cruzadas, es decir, utilizando dos de los campos. Los resultados de la búsqueda se muestran en forma de lista, igual que en venta artículos.

Nota1: Para realizar todas las opciones de introducción, modificación y eliminación de datos se pueden realizar con el programa MySQL Query Browser, el problema de modificar y eliminar datos con este programa es que no se realizan las comprobaciones pertinentes, por lo que se podrían eliminar datos en uso.

Nota2: No existe la función eliminar tamaño ya que siempre como mínimo deben estar las existentes de inicio.













4.2. Web

Con el rediseño de la página Web se ha pretendido hacerla más atractiva visualmente y más fácilmente navegable, ya que con el antiguo diseño, por ejemplo, el buscador de artículos no se veía claramente, cambiaba de sitio según la parte de la Web en la que nos encontrásemos, no se veía claramente donde dar de alta el usuario...La usabilidad de la página Web era bastante mala por lo que alguien que no estuviese realmente familiarizado o interesado en algo concreto de la página Web no accedía a ella.

Además la actualización ha sido casi nula durante los 3 últimos años, ya que en un principio la página Web sólo se iba a destinar a la venta de vinilos y CDs, por lo que no hacía falta un gran mantenimiento. Con la expansión de la empresa se ha hecho necesaria una ampliación de la página Web de manera que en un solo portal se pueda tener acceso a todo lo que la empresa quiere ofrecer.

Antes de llegar al diseño final que se optó para hacer el Web se pensaron diversas posibilidades. Algunas de ellas fueron:

Nuevo cliente(link) <input type="text"/> user <input type="password"/> passwd		FLASH	cesta compra <input type="text"/> buscador1 <input type="text"/> buscador2
Novedades Reposiciones Ofertas Trivial Rec Merchandisin Bookings Fiestas Links Presentación Info. Esto siempre	FLYER SIGUIENTE FIESTA CON INFOR DE LA MISMA, O MONTAGOOD, O ANIVERSARIO TRIVIAL...		novedades, siempre moviendose parriba y siempre visible.
Aquí merchandaisin debería ir cambiando cuando se clique en algún link			

inicio novedades Reposiciones	Trivialmusik Rec. Links Presentación información
OFERTAS	MERCHANDISING
     	     
<p>Contenido de Sección</p> <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. In in nisi. Praesent semper tempus nisi. Sed convallis mattis enim. Cras egetas, nisi id volutpat viverra, nisi pede cursus nisi, nec ornare una libero eu lorem. Suspendisse potenti. Cras lorem. Aenean at pede in ligula pretium volutpat. Fusce suscipit. Praesent eget ligula in leo consectetur eleifend. Nulla nec erat. Cras convallis pretium sem.</p> <p>Suspendisse nonummy felis ut tortor. Pellentesque elementum, neque sed bibendum feugiat, sem nunc venenatis lectus, ut bibendum quam lectus imperdiet ipsum. Proin suscipit, odio vehicula laoreet suscipit, turpis metus sagittis dolor, bibendum convallis neque dolor sed urna. Praesent nec quam. Integer ullamcorper sodales ipsum. Proin in ligula quis velit facilisis pellentesque. Cras hendrerit risus ac tortor varius molestie. Maecenas bibendum mi luctus tellus. Donec non est nec turpis sagittis tempus. Duis bibendum, diam cursus varius pulvinar, mi metus posuere tellus, in convallis quam diam id magna. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Sed tincidunt, mi ut congue nonummy, mi eros feugiat felis, a congue pede neque sollicitudin enim. Nunc condimentum, urna vitae condimentum tempor, urna libero posuere lectus, a imperdiet elit diam quis magna. Ut malesuada nisi id felis. Suspendisse in felis ut odio venenatis laoreet. Vestibulum elementum pulvinar diam. Sed vel arcu. Donec a ipsum.</p> <p>Praesent venenatis erat quis ligula. Aliquam adipiscing ipsum ac nibh. Curabitur nec augue ac dolor consectetur commodo. Nam pretium mi in ligula. Pellentesque pulvinar imperdiet mi. Suspendisse viverra. Nulla dignissim. Proin nonummy leo eget lacus. Cras venenatis euismod neque. Donec sit amet orci. Aliquam faucibus. Aenean tristique, ante sit amet blandit imperdiet, massa mauris vestibulum</p>	

5. Pruebas

La fase de prueba se puede ver como una fase destructiva del desarrollo del software donde se intenta encontrar casos de prueba que demuestre que hemos cometido errores.

Es imposible desarrollar el software sin cometer errores en las fases de especificación, diseño y codificación, por esta razón no se puede prescindir de la fase de prueba.

Para el desarrollo de la fase de prueba primero se deben establecer una serie de estrategias de prueba. Debemos tener claros que es imposible hacer una prueba exhaustiva y asegurar que el programa funciona al 100%. La fase de prueba es una de las actividades para asegurar la calidad del software.

Los principales objetivos de la realización de diferentes pruebas son:

- Examinar al máximo posible los posibles errores de código.
- Encontrar errores en el mínimo tiempo posible y con el mínimo esfuerzo.
- Comprobar que se logran todos los requerimientos especificados.

Por lo tanto se puede decir que la prueba es eficiente si cumple los siguientes objetivos:

- Un caso de prueba es bueno si tiene una alta probabilidad de detectar un error nuevo.
- La prueba tiene éxito si se encuentran errores no si no encuentra.
- La prueba no puede asegurar la ausencia de defectos, sólo demuestra que hay.
- Es encontrar errores contra más graves mejor, todo esto en el menor tiempo posible y en el menor esfuerzo.

En la estrategia de prueba se definen en que orden se prueban los módulos y funciones, cuanto tiempo durará la fase de prueba, que recursos utilizarán, etc. La fase de prueba se encuentra dentro del proceso de Verificación y Validación. La verificación es el proceso que asegura que el software esta implementado correctamente la función específica. El proceso de validación asegura que el software construido se ajusta a los requerimientos específicos del cliente.

Se puede afirmar que la fase de prueba no es un factor determinante de la calidad del software ya que la fase de prueba no resuelve errores, sólo los detecta. Para asegurarse de la calidad de un software (SQA) debemos hacer un seguimiento del proceso de construcción para que esta sea eficaz. Si se lleva a cabo al final del proceso de desarrollo del software el coste de los errores crece a medida que avanza. Muchos de los estudios realizados han demostrado que entre el 50% y el 65% de los errores se producen durante la fase de diseño.

Podemos decir que la fase de prueba no termina nunca, ya que no existe forma de demostrar que ya se han hecho pruebas suficientes para justificar que funciona correctamente. La prueba terminará cuando no tengamos más tiempo o se haya terminado el presupuesto.

Lo que garantiza la calidad del software (SQA) es buscar errores durante la especificación, análisis y diseño. Para garantizar la calidad del software se deben

realizar más procesos a parte de la fase de prueba. Los procesos que garantizan la calidad del software son:

- Revisiones Técnicas Formales: Es un mecanismo lógico de control de calidad del software. Se puede definir como actividades realizadas durante todo el proceso de desarrollo para encontrar errores. Buscan errores sin ejecutar el código, sabiendo que al detectar los errores lo más pronto posible el coste de la reparación es menor. Siempre las llevan a cabo un grupo de personas expertas en el tema a revisar.
- Estándares de desarrollo: Definen las diferentes características de tipos de productos de software y los procesos de Ingeniería del software. Los estándares describen las mejores prácticas del trabajo y seguirlos evita la repetición de errores pasados.
- Métricas del software: Medida cuantitativa del grado en que un sistema del software, componente o proceso posee una característica.
- Gestión de la configuración: Es el proceso de identificar y definir los elementos de un sistema. También controla los cambios de estos elementos durante todo el ciclo de vida y registra e informa sobre el estado de los elementos y peticiones de cambios.
- Prueba de caja blanca: Prueba si el software funciona correctamente. La prueba no es exhaustiva.
- Prueba de caja negra: Los casos de prueba se diseñan considerando exclusivamente las entradas y salidas del sistema, sin preocuparse por la estructura interna del mismo.

5.1 Pruebas de caja blanca

Basadas en demostrar que cada función implementada es totalmente operativa, buscando errores para cada función que implementa. Estas pruebas se centran en examinar los diferentes caminos de ejecución de una función, se procura garantizar:

La ejecución de al menos una vez de todas las líneas de código.
La ejecución de todas las decisiones lógicas en ambas posibilidades, verdaderas y falsas.
Todos los bucles son ejecutados hasta en los casos extremos.
Todas las estructuras de datos son utilizadas

Puede parecer que todas estas pruebas son suficientes para llevar a cabo un software 100% correcto, sin embargo para asegurar esto deberíamos definir todos los caminos lógicos y así diseñar casos de prueba que los ejecuten y evalúen.

Si pretendemos hacer esto en la práctica es imposible, sería muy compleja la búsqueda y ejecución de todos los caminos lógicos posibles. Por esta razón debemos escoger y ejecutar los caminos más importantes o los suficientes para cubrir todas las sentencias, utilizando las pruebas de caja blanca como los caminos básicos, la prueba de bucles, etc.

5.2 Pruebas de caja negra

Basadas en el conocimiento del funcionamiento del software e intentando probar que todos los elementos de desarrollo encajan, centrándose en el cumplimiento de los requerimientos e ignorando la estructura lógica interna de la aplicación.

Las pruebas de caja negra complementan a las pruebas de caja blanca, e intentan buscar los siguientes tipos de errores:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o acceso a la base de datos externas.
- Errores de rendimiento.
- Errores de inicialización y finalización.

5.3 Niveles de prueba

Considerando las pruebas de caja blanca y caja negra los diferentes niveles de prueba que se tienen que realizar en el desarrollo del software son los siguientes:

- Pruebas de unidad: Son aquellas en los módulos de la aplicación se prueban por separado. Estas pruebas, normalmente, se van realizando a medida que se va codificando la aplicación. Una vez finalizados los diferentes apartados de la aplicación se ponen en conjunto para formar el sistema. Son realizadas por el equipo de desarrollo.
- Pruebas de integración: El objetivo principal de estas pruebas es detectar errores en el diseño de la aplicación. Se limita a comprobar la interacción entre cada uno de los módulos de la aplicación y la compatibilidad de los datos que se intercambian. Al realizar una aplicación y una página Web que comparten información este tipo de pruebas son realmente necesarias ya que cualquier error en la aplicación afecta a la página Web.
- Pruebas de sistema: Son pruebas para comprobar el software ya incorporado con los elementos que forman el sistema, ya sean hardware o software.
- Pruebas de aceptación: Se realizan para demostrar al cliente la operatividad del sistema con datos y tiempo real.

5.4 Pruebas realizadas en la aplicación

Han sido muchas las pruebas realizadas para la comprobación del buen funcionamiento de la aplicación:

1. Comprobación del acceso de los usuarios al sistema, tanto al gestor como a la Web.
2. Comprobación de los datos (formato esperado y si son reales o no), inserción de los mismos en la base de datos conjuntamente con la

subida de archivos (imagen y audio) así como el envío de un e-mail cuando un cliente se registraba en la Web.

3. Comprobación a la hora de generar listados en la aplicación durante un periodo de tiempo para la tienda: compras, ventas, pérdidas...
4. Mostrar información de acciones realizadas en el gestor (log)
5. Comprobación de la correcta visualización de los listados de artículos cuando el cliente hace una búsqueda sea del tipo que sea.
6. Generación correcta de las facturas.
7. Comprobación de todas las funcionalidades, buscadores por palabra clave, búsqueda entre períodos de tiempo así como eliminar y recuperar elementos.
8. Comprobación de usuario existente y eliminación de la base de datos.
9. Redireccionamientos correctos.
10. A la hora de recuperar contraseñas comprobación de la existencia de la dirección electrónica introducida y generación de un e-mail con una nueva clave de acceso, así como modificación en la base de datos.
11. Comprobación de mostrar diferentes tipos de precios dependiendo de que tipo de cliente se trata.
12. Comprobación de la correcta modificación de datos en la base de datos.
13. Correcta visualización de posibles mensajes de error.
14. Comprobación de la correcta codificación de los datos de carácter personal introducidos en la base de datos.

6. Objetivos conseguidos

A la finalización del proyecto los objetivos conseguidos son:

- Creación de un gestor de stock capaz de:
 - Controlar las entradas y salidas de artículos en la empresa.
 - Proporcionar datos fiables sobre la entrada y salida de artículos.
 - Facilitar y aumentar la venta de artículos in situ.
 - Aumento de la rapidez a la hora de tratar ventas on-line.
 - Facilitar la entrada y el tratamiento de artículos en la base de datos.
 - Conocer las pérdidas reales.
 - Obtener listados de ventas, compras...que permiten conocer hacia donde y como varia el mercado.
 - Hacer copias de seguridad de la base de datos y ser capaz de recuperarlas.
- Modificación completa de la página Web, haciéndola:
 - Más usable, correcto posicionamiento todo el rato de los menús.
 - Más atractiva, haciendo que sea mucho más agradable el navegar por ella, y por lo tanto aumentando las ventas.
 - Más segura, se han codificado todos los campos con datos personales así como claves, de manera que sea casi imposible poder acceder a ellos. También se ha tratado la entrada de datos a la base de datos evitando inyecciones SQL y otro tipo de maneras de poder reventar la página.

Con todo esto se puede comprobar que se han cumplido casi todos los objetivos que se habían planteado antes de realizar este proyecto como son el realizar una aplicación que facilite el trabajo en la tienda así como el ampliar el mercado. Esto se ha conseguido informatizando todo el proceso de entrada y salida de artículos (control del stock), lo que a su vez facilita el saber que artículos tienen más rotación (listados de artículos). También se ha mejorado la presentación visual de la tienda y la seguridad de la misma, haciéndola mucho más atractiva y agradable a la hora de visitarla, lo que comportará un aumento de los beneficios y una expansión de la empresa.

En un principio se había pensado en hacer una aplicación con más utilidades y que englobase otras características como el "buqueo" (contratación de artistas representados) y el anuncio de fiestas. Pero una vez empezado el proyecto y con muchas ideas en mente se vio claramente que no se iba a disponer del tiempo suficiente para realizar todas, por lo que se acordó limitarlo, dejando ciertas características para un futuro cercano (posibles ampliaciones) y encargar otras a otros diseñadores Web (trivial bookings). Por todo ello se ha realizado una aplicación con una interfaz simple y a una programación bien comentada, de manera que en cualquier momento y cualquier persona con unos conocimientos mínimos pueda implementarlas.

6.1. Posibles ampliaciones

6.1.1. Contratos

En un futuro se pretende implementar una función que permita generar contratos de los representados en formato PDF. Actualmente esto se realiza manualmente, cada vez que hay que crear un contrato se deben introducir todos los campos en un archivo DOC y luego pasarlo a formato PDF.

El cliente deberá rellenar un formulario de formato Excel. Una vez se tenga el archivo el programa debe permitir cargar los datos del promotor del documento Excel o bien de la base de datos "Contratos". El programa debe permitir modificar los campos, ya que se puede haber introducido alguno mal, y por lo tanto también debe modificar el archivo Excel.

Con esta parte del programa se pretende agilizar todo el tema de contrataciones.

6.1.2. Inventariado

El programa deberá hacer un backup de la base de datos e imprimir una lista con todos los artículos que posee la tienda. Una vez impreso deberá preguntar si se desea continuar. Si se pulsa no finaliza el inventariado, sin embargo si se pulsa si el programa recorre la base de datos, y para cada referencia con cantidad diferente a cero nos pregunta cuantas poseemos, en caso de no coincidir se graba en un archivo de texto el artículo y la cantidad de los faltantes, de esta manera se pueden controlar los artículos extraviados.

6.1.3. Impresión etiquetas

Se introducirá una referencia y la cantidad de etiquetas a imprimir. Para poder realizar esta acción la empresa deberá comprar una impresora capaz de imprimir bandas magnéticas (código de barras) y un escáner capaz de leer los códigos de barras.

Se deberá crear un pequeño módulo capaz de crear códigos de barras únicos para cada producto, la manera de hacerlos será con el número de referencia, ya que es único para cada producto y por lo tanto el código será diferente también.

Se deberá adaptar el programa de manera que introduzca automáticamente el número de referencia a la hora de hacer un cobro.

6.1.4. Restock

Se pretende crear un apartado restock en el que el cliente puede pedir discos que en este momento no están disponibles. Cuando esos discos vuelvan a estar disponibles se avisa al cliente enviándole un e-mail y/o se le introduce en su carro de la compra, a este tipo de venta se le llama *Back office*.

6.1.5. Listado más vendidos

En la página principal, en la parte inferior se insertarán dos top diez, uno con las ventas del mes y otro con las ventas de la historia de la tienda.

6.1.6. Actualización de pedidos

Se pretende crear una función de manera que si un cliente tiene un artículo en su cesta de la compra que está agotado se le quite de la cesta y se le pase al apartado restock. Cuando esto suceda y se conecte a la Web se le avise mediante un mensaje (*pop-up*) en pantalla.

7. CONCLUSIONES

Internet es un sistema de comunicaciones de alcance mundial, barato, fiable y sencillo de utilizar. Hoy día, Internet cuenta con más de 400 millones de usuarios en todo el mundo y el número sigue aumentando constantemente a causa de la facilidad que brinda Internet para obtener información rápida y barata desde la comodidad de casa o bien desde la oficina. Este número creciente de usuarios pueden ser clientes potenciales para una empresa o negocio. En Internet, cada día también aumenta la cantidad de productos y servicios que se comercializan mediante este medio, facilitando conseguir mejores precios y nuevos proveedores, mejorando así los márgenes de utilidad.

Todo esto hace que cualquier empresa o entidad que quiera darse a conocer o expandirse prácticamente está obligada a tener su parcela en el mundo Web.

Partíamos de una metodología particular (manual y poco eficiente), por lo que era poco funcional y mucho menos óptima. Con este proyecto se ha conseguido mejorar las funcionalidades, ofrecer nuevas y mejorar considerablemente el rendimiento de la empresa. También se han conseguido otros objetivos como mejorar la competitividad de la empresa, minimizar los errores de gestión, mejorar la imagen de la empresa de cara nuevos clientes y hacia otras empresas.

También hay que hablar del mantenimiento de la aplicación así como el correcto cumplimiento de la ley de protección de datos de carácter personal (LOPD 15/1999). Esto será responsabilidad de los Srs. Víctor Peral y Jordi Castilla, dueños de la empresa.

En caso de realizarse alguna ampliación será realizada por mí pero no de forma gratuita como ha sido realizado este proyecto.

Por último comentar que con la realización del proyecto he podido comprobar lo difícil que es que un cliente llegue a comprender lo que se le pide y que te lo de, por lo que la creación de la aplicación se ha finalizado con retraso, todo esto partiendo de la base que poseía los conocimientos suficientes y necesarios para facilitar la tarea al haber estado trabajando en la tienda durante un año. Además de poder comprobar que cuando empiezas a hacer un trabajo de estas dimensiones es muy difícil abarcar todo lo que se quiere hacer, en definitiva una mala estimación de tiempos por la falta de conocimiento en trabajos de este tipo. A pesar de todo esto se puede decir que ha sido un trabajo muy instructivo y que me va a servir para poder ver las cosas desde otra perspectiva y así poder hacer valoraciones mucho más correctas y exactas.

8. BIBLIOGRAFÍA

Recursos impresos:

- [1] TRIGO ARANDA, Vicente; Creación de páginas Web online, Anaya Multimedia, Madrid, 2001
- [2] RATSCHILLER, Tobias; Creación de aplicaciones Web con PHP 4, Prentice Hall, 2001
- [3] EÑA DE SAN ANTONIO, Óscar; Dreamweaver MX 2004, Guía práctica para usuarios, Ediciones Anaya Multimedia, Madrid, 2004.
- [4] NT, Allan; Desarrollo Web con PHP y Dreamweaver MX 2004, Ediciones Anaya Multimedia, Madrid, 2005
- [5] GALLEGO VÁZQUEZ, José Antonio; Desarrollo Web con PHP y MySQL, Ediciones Anaya Multimedia, 2003
- [6] GOTO, Kelly; Rediseño y desarrollo de sitios Web, Anaya Multimedia, Madrid, 2005
- [7] Apuntes de la asignatura Interacción Humana con los Ordenadores (UOC) 2006/07

Recursos on-line:

Categorías	Web	Fecha
Bases de datos		
	http://www.mysql.com/documentation/index.html	15.01.2009
	http://www.php-es.com/ref.mysql.html	04.2009 – 06.2009
	http://www.mygnet.com/manuales/mysql/	15.01.2009
	http://www3.uji.es/~mmarques/f47/apun/node83.html	10.06.2009
HTML		
	http://www.webestilo.com/html/	03.2009
	http://www.alzado.org/articulo.php?id_art=57	03.2009
PHP		
	http://php.net	03.2009 – 06.2009
	http://www.manualdephp.com/	04.2009
	http://www.sortea2.com/blog/?p=66	13.05.2009
	http://www.colewebs.org/spip.php?article24&esqueto=bones_kubrick	26.06.2009
Javascript		

	http://www.mundojavascript.com/	03.2009 – 05.2009
	http://www.webestilo.com/javascript/	03.2009 – 05.2009
	http://www.lawebera.es/manuales/javascript/caracteristicas.php	03.2009
AJAX		
	http://www.desarrolloweb.com	03.2009 – 06.2009
	http://www.ribosomatic.com/categoria/ajax/	04.2009
Servidores		
	http://web.iespana.es	20.07.2009
Diversos temas		
	http://www.forosdelweb.com	01.2009 – 06.2009
	http://www.programacionweb.net/	01.2009 – 06.2009
	http://www.elguruprogramador.com.ar/foros	01.2009 – 06.2009
	http://www.solomanuales.org/	02.2009
	http://soporte.miarroba.com	02.2009 – 05.2009

Software utilizado:

- Dreamweaver CS3
- WAMP5 Versión 2.0
- Office 2003
- Microsoft Office Visio 2003
- Magic Draw v.16

